Bell
Lewis

HAWKATARI ②

THE CEDAR RAPIDS & IOWA CITY USER'S GROUP

February 1987   Page 1

Next meeting:
February 17, 1987 7:00PM
Cedar Terrace Mobile Home
Estates Recreation Room
1.7 miles West of
Intersection of Edgewood &
16th Avenue on 16th
Avenue

Equipment:
   800XL
   1050 Drive
   Color Monitor

Minutes of January Meeting
    The January 20, 1987 meeting of HawkAtari was called to order at 7:03PM by President Jim Koehn. There were 21 people present.

New Business: The club has purchased a monitor for it's computer system from K-Mart. Jim had no further details on a club purchase of monitors from K-Mart as his contact there did not contact him as promised before the meeting.
    Ron Grassi mentioned details on the Iowa City ST/Amiga computer club meeting on January 26th at 7:30 PM. Contact Dan Ruby at the Parallel Port in Iowa City for further details. Phone 354-1097.

Welcome to new club members:
Donald Crow
David Cawthorne
Richard Gott

AMAC for RAMDISK USE
Paul Maternowski

    The use of a RamDisk with AMAC (Atari MACro assembler) is not possible without a small patch of 1 byte in the file. The AMAC assembler tests the drive name for an extension > 4 being a error. It does this with a CMP #5, BCC sequence.
    To correct this search for the sequence `$C9,$05 on the disk using Disk Doctor or another disk utility supporting this type of search in DOS mode. This sequence is the only one in the AMAC file. Replace the $05 with $09. The C9 05 tests for the disk drive designator being less than 5. Changing it to a 9 allows a drive designation up to 8. Of course always do this on a copy not the original!! However on a copy the copy protection will have had to be removed also for the file to work.

Newsletter Brief: XL/XE Power Supplies
    Reprinted form Abes ACE Compliments of Milatari

    Discussion with schematics of the 4 types of XL/XE power supplies. Will be available at February meeting for interested parties.

AtariWriter+ Printer Drivers
Excerpt from Computah Author: Jolene Morris

    Do you enjoy using the AtariWriter+ program but find it doesn't support your particular brand of printer? First of all, check with your printer salesman and see if your printer is compatible withanother printer. For example, a CITIZEN printer is compatible with an EPSON FX-80/85 so you can tell AtariWriter+ that you have an EPSON when you get ready to print. If your'e printer isn't compatible with any of the listed printers, you will have to build your own printer driver. I have a PROWRITER printer and had to build a driver. Here is how you do it. (I will be using the PROWRITER as an example when I explain how to make a printer driver but you can make a driver for any brand printer).
    First of all, you need three things: your copy of the AtariWriter+ disk, a formatted disk and a copy of your printer instruction manual. Someplace in your printer manual there

will be a page with a printer control chart on it. It usually tells the function, the ESC code, the decimal code and the hexadecimal code to get these functions. Locate that chart. You must use the part of the chart that gives the decimal numbers for the functions.

Now insert the AW+ disk in your turned on drive and close the latch. (ATARI 800 owners put in your BASIC cartridge). Hold down the SELECT key and turn on the computer. Keep the SELECT key down until you see the CUSTOM PRINTER title page. Now hit the SPACEBAR and a list of printer functions will appear on your screen. You will see an arrow pointing to the first function (INITIALIZE EVERY LINE). Now you just need to enter the numbers for each of those functions. Let's do the first few together step-by-step:

1. With the arrow on INITIALIZE EVERY LINE, press RETURN. Type a 27, press RETURN. Type a 91, and then press RETURN twice. These are the printer codes taken from the printer manual.

2. Move the arrow to LINEFEED AND CARRAIGE RETURN (using the down arrow key) and press RETURN. Type the follwing: 27,RETURN. 93,RETURN. 27, RETURN. 65,RETURN. 27,RETURN. 102, RETURN. 13 and RETURN twice.

3. Move the arrow to UNDERLINE OFF and press RETURN. Type: 27,RETURN. 89 then RETURN twice.

4. Try the rest on your own.

5. To do the TYPE FONTS:
  Set up the type fonts as above.
6. Press S to save your custom printer file. Insert your formatted disk in the disk drive then type a filename such as: D1:PROWRITER and RETURN.

7. To return to AW+ press R and it will ask you if you want to return to AW+. Press Y and you will be asked to insert your AW+ disk and press START.

8. From then on when you want to print from AW+'s main menu, press P, then choose option H for other printers. Then choose option O for custom printer drivers. When it asks for a filename tell it the filename you typed in when you saved your custom printer driver and press RETURN.

Disable Basic Easily
Reprinted from MONITOR Issue 11 the U.K. ATARI Compliments of SLCC Journal

For those of you who load binary files with an AUTORUN.SYS menu loader, the following procedure will allow you to boot without holding down the option key.

Boot a DOS 2.0S disk while holding down the OPTION key (for the last time). Then place your AUTORUN.SYS loader file in the drive. Now hit the 'E' option to rename the loader (e.g.: AUTORUN.SYS,AUTORUN.OLD). Next hit the 'K' option to binary save AUTORUN.SYS at a starting address of D301 and an ending address of D301 (e.g.: AUTORUN.SYS,D301,D301). This places an 'FF' in decimal position 54017 which tells the OS to disable BASIC. Finally hit the 'C' option to copy with append AUTORUN.OLD to the just saved AUTORUN.SYS file (e.g.: AUTORUN.OLD,AUTORUN.SYS/A).

Editor's note: This should work with other derivatives of DOS 2.0 on XL & XE computers.

Connecting a RGB Monitor to a ST

---

outstanding 8-bit program on Astromony. This type program really lets the 130XE show her stuff as compaired to the other 8-bit machines. The only thing this program won't do is let you descover a new Star, planet, comet,etc. It can do several events that real telescopes can't do; like look into the sky any where, any day and at any time. Call Mike at 273-1185 for more info or HELP. The program costs from $23 to $30. depending where you but it.
Well we wound down a little after noon; and as you can see it was on 8-bit meeting. Rather boring for ST'ers. Listening over Steves shoulder he indicated that he would like to see future meetings something like 3/5ths ST vs 2/5ths 8-bit.
Meeting adjourned at 12:02PM

KRYSA
sectry
N.W.P.A.C.

From the pages of P.A.C.E. WORLD
  November 1986

By Steve Fishbein(with great assistance from Jeff
Fishbein and Chester Marks)

N W PAC
2-87
ATARIWRITER +
A RAMDISK CONVERSION

Working with various word processors it had appeared that only two were making use of the RAMDISK capabilities of the 130XE, Antic's Word Magic and the Ilent WP. My son Jeff, an avid Atariwriter+ user, was lamenting the lack of a RAMDISK and the convenience it would offer in using our Atari computers at the annual Virginia Duals Collegiate Wrestling Tournament.

I had recently used the Chipmunk program to create an unprotected version of Atariwriter+ and it occurred to me that we might be abel to use this copy and add RAMDISK capability.

Realizing that the program only allowed indexing of drives 1 and 2, we were pleasantly surprised to find that just substituting DOS 2.5 for the original DOS and adding the ramdisk file, Atariwriter did allow saving and retrieving to D8, but with no provision to index the ramdisk.

We kept this version as an alternate, but then went forward in order to come up with a ramdisk using drive two. A telephone call to Chester Marks, who had set up my 850 Express with Sparta Dos, brought forth the needed information how to set up Sparta Dos and change the ramdisk from Drive 8 to Drive 2.

This version did permit indexing of the ramdisk, eliminated ④ the need for holding the OPTION key when bootng the program and resulted in one other very big PLUS - an Atariwritr much, much faster than the original.

The new revision, however, was not perfect. We discovered that memory banks 2 and 3 could not be used with the Sparta Dos version. Attempts to save files longer than the capacity of Bank One crashed the file. There was no problem saving and retrieving to D8 with the DOS 2.5 version.

The solution was to use the Sparta DOS version for short files, not easily recalled, and the DOS 2.5 version for a long file.

I concluded the experimentation by making one more version - an Atariwriter + using Sparta DOS instead of the original DOS. This gave me an Atariwriter which had the high speed and didn't require holding the OPTION key while booting.

Sevac Minutes

The Jan 17th meeting of the SouthEast Valley Atari Connection was called to order at 10:10 am by Tim Barr.

After reading the minutes of the December meeting, Lee Whiteside gave the Treasurer's report. Current cash in the account is at $1200. A vote was taken to approve paying bills totaling around $300 to NWPAC for newsletter costs and half of the BBS phone line costs.

After an update on the 8 bit and ST public domain libraries, Lee Whiteside gave a report on the Winter Consumer Electronics show held in early January in Las Vegas (reports should be elsewhere in the newsletter).

Next the elections for club officers was held. Since there was only one candidate per office, a voice vote was held with the following unanimous results.

President   : Tim Barr
Vice-President : Martin Young
Secretary   : Don Daniels
Treasurer   : Lee Whiteside

After a break, Martin Young gave demos of the GEM version of VIP Professional and of Publishing Partner, both for the ST. The meeting adjourned at 12:10 pm.

The next meeting of SEVAC will be on Saturday February 21st at 1 pm in the Mesa Library, 64 E 1st St.

COLDSTARTING CAN HELP YOUR COMPUTER LIVE A LONGER LIFE

3-87
Wayne Habberstad - LA-ACE

Are you zapping your computer to death? Whenever you turn on your computer, a surge of power shocks its internal components, especially its power supply. While the computer can stand repeated "zaps" over a few years, they do eventually do the poor computer in.

Turning on and off a computer is inevitable; but you can minimize it by fooling the computer into thinking it has just been turned on. When an Atari 8 bit computer is turned on it clears all of its RAM memory (except the extra RAM that lies under the ROM in XL/XE computers and the extra RAM in the 130XE) and sets itself up ready to accept programs.

The routine in the Atari's operating system that does this chore, called coldstarting the computer (as opposed to warmstarting which is simply the resetting of the computer, usually by the RESET key) can be called by you from BASIC or DOS when warmstarting the computer is not enough to clear an unwanted program from memory, or the garbage created by it. To call the routine from BASIC, use this statement: A=USR(58487). When you press the return key the computer will blank out the display and act as though it had just been turned on.

With an XL/XE computer you can coldstart it by first using the statement BYE (or the abbreviation B.), then pressing the RESET key when the self test menu appears. The RESET key coldstarts the computer in this instance because the self test routine had inserted a one into location 580, which is checked when the RESET key is pressed; any other number tells it to coldstart. You can use the POKE statement in BASIC to set location 580: POKE 580,1. Then the RESET key will coldstart the computer when pressed.

If you are not in BASIC, this is how to coldstart the computer in ATARI DOS: use menu option M (run address), then use $E477 (the hexadecimal equivalent of 58487) as the address.

Having to clear out old programs, and their results, is not the only thing to tempt you into turning off, and on, your computer. Getting boot errors on your screen after you put the wrong disk in the drive may lead you to think that turning off the computer (or the disk drive) is the only way to stop the drive from spinning the disk inside. And pressing the RESET key is not enough as the

computer will again uselessly try to boot up with the wring disk. The si the way around it: hold the START key down and press the RESET key. The computer will think that you want it to boot up from a cassette instead of a disk (the beep that you will hear is its signal to you to ready a cassette [;auer). The disk drive will stop spinning the disk. After you replace the disk in the drive press the RESET key. The computer will forget about trying to boot from a cassette and will boot from the disk.

Your Atari is the best 8 bit computer ever made. It deserves to be well taken care of.

FOR 130XE OWNERS ONLY

For 130 XE owners there is a bonus if they coldstart their computers: programs and data on the ramdisk are safe from the coldstart process as long as there is no RAMDISK.COM on the boot disk. As mentioned earlier the coldstart routine in the operating system does not touch the extra 64K in the 130XE. But the RAMDISK.COM will reformat the ramdisk and wipe out everything on it. As the RAMDISK.COM is needed to set up the ramdisk when the computer is turned on, 130 XE owners should keep two copies of their boot disks: one with the RAMDISK.COM to use when turning on their computers, and the other without it to use when merely coldstarting them.

After a 130XE is coldstarted without a RAMDISK.COM on the boot disk, the ramdisk can be read from and written to normally except for one thing: if you try to go to DOS from BASIC then the computer will attempt to load the DUP.SYS file from Drive 1 instead of from the ramdisk. The way to remedy this is to POKE 5439,56. This will tell the computer to load DUP.SYS from the ramdisk.

ACTION!/6502 Assembly Class

Mark Manyen LA-ACE

As was announced in the last LAACE Newsletter, I will be giving a class once a month at the Granada Hills High School dealing with the ACTION! language from OSS inc., as well as 6502 Assembly Language. The first meeting of this class will be Wednesday, March 4th at 7:30 P.M. If you intend to be part of this class, please call me to sign up at (213) 471-2586 after 6:00 P.M. weekdays or see me at the March meeting. If this time is bad for you, please let me know as I am sure that arrangements for an alternate meeting place and time can be found if enough people cannot make this time. If the February LAACE meeting was any indication of the response I can expect to this class, THERE WILL BE NO CLASS. So, if you are interested, make sure you let me know.

THE 1050 DUPLICATOR - REV. 3.0
You Can't Always Believe What You Read

Review by Rick Holtzhauer

Duplicating Technologies (DTI) has recently released it's long awaited Rev 3.0 for the 1050 Duplicator. According to Mike Carney of DTI, so advanced would this revision be that all Happy owners would "... have to throw their Happy Enhancement in the garbage... " (See CN Jul/Aug '85).

Hmmmmm? Would Happy Computers be facing the first serious challenge to it's total domination in the disk backup/high performance drive field?

Hardly! I found out very quickly you can't always believe what you read, as the Duplicator has in fact turned out to be perhaps the biggest lemon ever foisted on the Atari world.

The software revision came on two disks, and many Duplicator owners also needed an EPROM upgrade (this despite the fact DTI's advertisements claim "no further hardware upgrades ever needed". (I'm on my second EPROM; I've heard of some users on their third.) I've limited the scope of this review to the three main functions of the new software/hardware package -- the backup program, the sector copier and the handling of skew-aligned programs.

SKEW ALIGNMENT

The skew program is a binary load file that will backup disks that employ a protection DTI terms as a "light skew", or a "track skew". A track skew is a disk format where the position of one track is directly relational to the track preceding and following it. For example, take the skew used on *F-15 Strike Eagle*. When the time comes for this program to go into it's protection check, the onboard timers are set to zero. Then, the program reads, say, sector one of track two, track three, four, five, and six. The value of the timer is loaded and saved, reset to zero, and the program then reads sector one of track six, track five, and so on back down to track two. The timer value is again loaded, then compared to the timer value previously loaded and saved. If these values are not very close, the program will crash. On a normally formatted Atari disk, the time to read the first set of sectors will be much longer than the second set (at least on my drive). Thus, a valid protection program.

The track skew program on Rev 3 reproduces this skew extremely well. DTI claims the reason a separate backup program for skew was necessary is because "... making a backup program of a skew disk is a much slower process than other types of protection ...." Not true. Though a skew backup does take quite a bit longer than a normal backup, the reason they made this a separate program is

because DTI still cannot grasp the basic and fundamental aspects of backing up a skew disk. They had to write a specific program to format a specific skew. They could not write a program that could analyze and duplicate all forms of skew. For instance, Electronic Art's *Pinball Construction Set* (copyright 1983) uses what DTI terms as a "heavy skew". The Duplicator STILL cannot back-up this skew format.

Let's examine a much more serious flaw in this module. Let's examine a sector-skewed disk, or, more precisely, a disk formatted on the Indus GT in the Synchromesh mode. A sector skew is the way individual sectors are arranged inside of a track. Because the Indus drive is capable of transferring data much quicker than the 1050, this sector arrangement is different than the stock 1050. Although the capability to reproduce this type skew was promised in Rev 3.0, the Duplicator still cannot duplicate an Indus/Synchromesh formatted disk and have the disk run at optimum speed on the Indus. Now, of course, this capability is promised by DTI in the yet to come Rev 4.0.

This lack of skew capabilities is interesting when one realizes that every software house in America can successfully protect their products from the Duplicator's normal back-up module by merely formatting their programs on an Indus drive in the Synchromesh mode.

ULTRA SPEED HANDLER

The Ultra Speed handler, when used in conjunction with the sector copier, is DTI's answer to Happy Computers Warp-Speed sector copier.

The operation of this handler is, at best, very awkward. To load it, you must first load the menu disk, and choose the handler. The drive loads the handler, then instructs you to insert the disk to boot in ultra speed, and a coldstart is performed after about a ten second delay.

The speed handler can also be renamed AUTORUN.SYS, then copied back to Dos 2.5. Your drive will then operate in the ultra speed mode. There are a few problems using this speed handler in a DOS environment, though. First of all, the speed handler is 90 sectors long (vs. the nine sectors, I believe, for Happy's Warp Speed Dos 2.0). After it loads, another separate Dos 2.5 disk must be inserted, because the program jumps to the coldstart routine.

When I finally got my handler installed and DOS 2.5 back on the screen, I found the handler refused to read my disk directory. This is documented as an "occasion-

al" problem in the Rev 3 manual. The solution? Merely hit the BREAK key. But in operation this glitch is much worse than I could have imagined. Four times out of five, I could not read the directory, and ended up pounding on the BREAK key in sheer frustration. Overall operation of the ultra speed handler is totally unacceptable, the bugs and glitches making it nothing short of useless.

When using the ultra speed handler with the sector copier, the program read the source disk at an extremely high speed, but it appeared to write very slowly. As an experiment I inserted a SPARTA DOS skew-formatted disk in my drive. The write time now increased significantly, over five seconds faster than the Happy can do on a 360 sector copy. This implies that DTI's Ultra Speed handler was not designed well for its own system. It requires a sector skewed disk to efficiently handle high speed I/O writes.

SECTOR COPIER

DTI makes some incredible statements regarding the sector copier: "OUR sector copier has many excellent features. One of the most important is this program will go through all errors". They wrote and informed Current Notes readers (July/86) that "Our almost completed sector copier can reproduce a deleted data mark in ultra speed".

The sector copier will not reproduce a deleted data mark, and the copier module does not even appear to be DTI's product. The program is actually "SCOPY 810" lease 10, written by Craig Chamberlain and copyrighted by Alliance Software in 1982. It was released through several users groups, this particular version by The Jersey Atari Computer Group. The only difference is that the programming credits and copyright notice have been dropped. This may be demonstrated by using a sector editor and reading the first sector of the "HCOPY" file of the speed handler disk.. On the Rev 3 disk it is sector 210. At location $3A, you will find the byte sequence "4C 74 E4". Change this to "6C 0A 00". Now, load the sector copier and behold, the original "SCOPY 810" copyright notice appears! Seems their "almost completed" sector copier is actually over four years old! It will not handle "enhanced" or double density disks. Who had anything but single density disk way back in '82?

REV 3 BACKUP

My original review of the 1050 Duplicator, version 2.1, appeared in the June '85 edition of CN. Although DTI's response to this article indicated they were then shipping software revision 2.5, it wasn't until five months later that I received my copy of 2.5 with an EPROM upgrade. I found the performance of 2.5 to be pretty bad -- not much better than Rev 2.1.

As Rev's 2.1 and 2.5 were so disappointing, I expected great things of Rev 3.0 if it was to live up to

its advance hype. It even promised the capability of backing up weak sectors, a format that Happy Computers, at the present time, cannot handle.

Alas, it was not to be. Copy capabilities of Ver. 3.0 are virtually nonexistent. It can handle a non-skewed, simple 18 or 19 sector track, which, incidentally, is nonexistent on any recent release. In theory, it may copy a 20 sector track by slowing the drive down to 270 rpm. It must slow down the drive to 270 rpm because the programmers at DTI apparently still lack the skill to manipulate the sector ID bytes so 20 sectors can be written at normal drive speed, as both the Happy and Archiver 1.0 (copyright: 1983) can do. The Duplicator can write a 21 sector track if the drive speed is set at 255 rpm, while the Happy and Archiver are set at 270 rpm.

But the ability to write a 20 sector track does not mean the ability to copy a 20 sector track, as the Duplicator so dramatically proved. I attempted to back up all my 20 sector disks, among them: Silent Service, Kennedy Approach, Lode Runners Rescue, Koronis Rift, and others. The Duplicator backed up nary a one. The program consistently wrote the wrong data in the wrong sectors, and lacked the ability to write a full, 128-byte sector with a CRC error (an Archiver fault, also). Amazingly, Rev 2.1 can back up Koronis Rift, but the Rev 3 could not.

The copy process is a good six minutes long, and there is no selective track copying; it's the entire disk or nothing.

The original Duplicator manual promises the backup program will run in the ultra speed mode. It does not. They also promise it will take advantage of the extra memory on a 130XE computer. It does not.

The reason this revision offered no improvement over Rev 2.5 has perturbed me greatly, and that's because Rev 3.0 IS Rev 2.5. DTI took the Rev 2.5 software, changed two bytes in it (Rev 2.5 to Rev 3.0), then released this as the Rev 3 "upgrade". Of course, DTI will claim the real upgrade was the EPROM. Wrong. According to the Rev 3 release letter, only "...some of our older customers will need the Rev 5 EPROM...". Which means quite a few Duplicator owners (like me) waited months for DTI to release software they already possessed.

MODULES

DTI has made some provisions for backing up software that the Rev 3 copy program cannot handle. These are copy "modules", and are loaded through a self-booting disk. The disk contains a list of ten programs that will backup disks that employ weak sectoring (Never Ending Story, Goonies) and the "trick" 20 sector format (Silent Service, Fight Night, Sargon III).

As an example, let's go through the backup procedure for "Silent Service". You are first instructed to copy

the program with the Rev 3 copy program. You then load up the module disk, and are presented with a menu containing the ten programs these modules can cover. I chose the appropriate module for Silent Service. After loading the module, you are instructed to insert the backup copy in the drive. After a few moments, the drive will do some writing, and the copy is done. The copy indeed booted and ran correctly.

However, as the backup program was loading little bells and whistles began going off in my head. Upon examination of the backup copy of Silent Service, I found out that the protection was not duplicated; the disk was broken — the PROTECTION CODE WAS REMOVED FROM THE PROGRAM! Future copies can be made from this backup copy using any sector copier. In fact, the Rev 3 copy program is not needed at all, just do a sector copy of the original program and load up and run the module program.

I ran the Silent Service module again. When the computer instructed me to place my backup copy in the drive, I instead inserted a freshly formatted, blank disk. After the program finished writing to my disk, I removed it and did a sector to sector comparison between this disk and the original Silent Service program. I found that sectors 37-38, 42-46, and 48-56 were the same.

Interesting, I thought. Since the original disk is not read when the backup is being created, the only other place this data could have come from is the Silent Service module. Which means DTI is mailing me a part of the copyrighted Silent Service program, without, I feel safe to say, Microprose's permission.

This method of backup is in stark contrast to recent advertisements in ANALOG and ANTIC for the Duplicator. According to the ads, the Duplicator will: "... reproduce any custom format or heavily copy guarded scheme... Custom formats will be read and in turn REPRODUCED on the backup copy disk... We are now doing the weak sector, eg. Never Ending Story".

These advertisements would lead a potential buyer to believe the Duplicator will successfully duplicate the weak sector protection on the backup disk, which is not true.

I must also wonder how long these copy modules will be around before DTI starts getting hammered by the companies that market the software these modules cover.

INNOVATION

DTI claims they plan to write many new and exciting programs that will operate only on a Duplicator. They have been promising this for a year now, and none have materialized. DTI proved incapable of even writing their own sector copier.

COMMENTS

The method that DTI has chosen to "backup" the programs it could not actually backup angers me. I would like to discuss the ethical and moral aspects of any company releasing a program such as this.

For about the last year and a half, we have seen much more sophisticated and determined methods by software companies to protect their investments. They have, finally and permanently, turned the tables on disk backup systems. No more will you be able to slap a program into your Happy and give it to your buddy.

Happy Computers' Rev 7 allowed backup of these disks, though they must be run on a Happy drive. And with this, it seems as if we have hit a truce in the protection wars. Happy Computers allows the user to backup his software; the software house knows these backups won't be floating all over Atariland inasmuch as a Happy enhanced drive is required to run the copies.

Now we have DTI releasing a revision that will UN-PROTECT programs on the market. Products like this pose the greatest threat to the Atari 8-bit line.

When the Happy drive was first released, the protection battle was on. A new protection scheme would come out, and the Happy would copy it. Back to the drawing board, where yet a new protection scheme was devised, and on and on. But at least for the software house, there was always hope the perfect format could be devised. Eventually, as explained above, this hope paid off. But Atari users paid heavily. Some companies, rather than get caught in this war, chose to abandon the Atari line. Atari versions of some programs were not released until a suitable protection format was devised.

Though the software houses eventually beat back the Happy challenge, they cannot beat back this challenge from DTI. It is virtually impossible to write the protection code that cannot be broken. To write this protection would probably take as much effort as the program itself took.

No doubt, Happy had the foresight to see that releasing a program similar to DTI's would not be beneficial to the Atari world. So the backups were designed to operate only on the Happy drive. A company called C.S.S. Software out of New York markets a similar called The Impossible. This program will take a protected disk and rewrite it without the protection. But this program would only operate if a special cartridge was inserted into the computer.

By adopting their approach to "backing up" heavily protected programs, isn't DTI actually advocating disk piracy? Let's take another look at what they're doing.

When Microprose released F-15 Strike Eagle, it employed a new protection twist. It required the user to

enter an access code for the game to operate correctly. These codes were scattered throughout the users manual. No manual, no play. You must purchase the program to get the manual, unless the manual was also illegally copied. Since the codes were simple, though, you could just write them down on a piece of paper.

Then along came *Silent Service*. Instead of using simple number-letter codes, the program instead drew four very similar ships on the screen. The user then had to compare these with ships scattered throughout the users manual, and choose a correct match. This is no piece of cake. To this day I am still entering the wrong code, even with the manual open in front of me.

But DTI, in addition to stripping away the disk protection, has eliminated the subroutine that calls for the authentication code to be entered. Now, what reason on earth would they have for doing this?

To further complicate matters, consider the process DTI used in creating these backup modules. In theory, you must have a Duplicator to use these backup modules.

When a backup module is chosen from the menu and after the file is loaded, the program then stuffs a $62 into location $302 (DCOMND). It then does a JSR to location $E459, with communication between the drive and the computer then taking place. The value $62 I think is a poll of some sort, like an "are you there?" request. Only a Duplicator responds to this disk command, no other Atari drive will. If the answer from the drive is positive, normal operation of the backup module resumes. If the answer is negative, the program aborts, and a "Duplicator needed to operate", or words to that effect, are printed on the screen.

So I had an idea. Instead of using the value $62, I substituted the value $53 instead. This value is the drive status request command, a command every Atari drive must respond to. As required, the drive did indeed respond. The backup module took this to be a positive response, and the program proceeded as it should, and the backup was completed as normal. Except I was using an Indus drive. Uh oh...

I next did a sector search of the entire backup module disk for the byte combination A0 62. This combination showed up eleven times. I found if I changed each of the $62 to a $53 and rewrote the sector back to the disk, these modules would operate on any drive. Meaning I could do a sector copy of *Silent Service* on my Indus, load up the copy module on the Indus, and have the Indus create a working copy. No $152.45 Duplicator needed!

I chose to include the above information in this review only after long thought, and then only because of the simplicity of finding it. Inside of five minutes. But since it was so simple for me, it was also simple for many other users. Now, how long do you think it'll take before these modules, allowing for operation on any

drive, to start appearing on bulletin boards all over the country?

SOLUTION

Though there may be many solutions, there is only one in my book. Atari users must clean their own house.

I've read until I am blue in the face letters and editorials condemning disk piracy. But it seems the same magazines that run these editorials and letters on one page, carry full page ads for the Duplicator on the other. This makes no sense. This is not a backup device. It's too easily adaptable to true pirating purposes.

Actions, however, speak louder than words. Seems to me that it's the responsibility of all Atari owners to write ANALOG, ANTIC, and COMPUTE! magazines, condemning this backup method and urging that all advertisements for the Duplicator be dropped until an alternative method of backup can be achieved. And to kill two birds with one stone, send a message to all software houses at the same time.

Failure of the Atari community to take any action and allow the Duplicator to flourish will also send a message to software houses, and to many hackers. I consider myself fairly good at breaking protected disk. Does that mean I, and many others, can get in on a piece of the action?

SUMMARY

Inferior in all respects. DTI has failed to deliver on virtually every single promise made for Rev 3, and instead delivered a very crude and amateurish attempt far outperformed by even the Archiver. It should not carry a $49.95 price tag, much less the $149.95 they ask for it. I do not like this company. I do not like this product.

ERRATA: My review of the Happy Rev 7 (CN OCT86) incorrectly stated that the Warp Speed DOS could be used on DOS 2.5. It cannot.

Interested readers may write to me at the following address: Rick Holtzhauer, NEESA, Port Hueneme, CA 93043

(XE Editor's Note: When Rick returned to his base in California after spending the Christmas holidays in Virginia, he found a letter from DTI awaiting him. DTI's letter announced that their Revision 4.0 was now available to 1050 Duplicator's owners at a cost of $13.95 plus shipping and handling. The letter states, in part, that

"... our Module creator is enabling us to get our revisions out much more quickly and efficiently. We now have the technology to backup any form of protection on

---

the market today. Even disks protected by hardware keys that must be present in the joystick ports offer no problems for the Duplicator.

The module program will remove the protection from the copyguarded disk allowing our customers to make backup copies of their valuable software and these backups you make can now be sent over the phone lines by modem...

We have pledged to our customers that whatever the software manufacturers DO, WE WILL UNDO ...

There are over 100 modules on this 4.0 Revision ..."

DTI's letter acknowledges at least one of the points Rick sets forth in his review. DTI has removed the protection code from their "backup" copies made using their "Module" system. The DTI disk containing the individual modules apparently contains original code, copyrighted by the targeted program's publisher, which is written to the backup copy. I'll leave the morality and legality of this approach to those more versed than I in the arcane convolutions of copyright law, but I suspect DTI will not go unchallenged.

Shortly after Rick's first article on the 1050 Duplicator appeared in CN last June, I received information that DTI had been shipping an unmarked copy of ICD's *SpartaDOS* with the initial Duplicator packages without authorization from ICD. I called Tom Harker of ICD and he confirmed DTI had been shipping unauthorized copies of *SpartaDOS*. He said he had reported that matter to the FBI, but that they refused to take action in such cases unless at least a half-million dollars in losses was involved. Harker said he then discussed the matter with his own attorneys who advised that it would cost at least $30,000 to pursue a copyright infringement charge against DTI. Harker said he threw up his hands in disgust, and gave up, but characterized DTI's practice as "outright theft".

It might be easy to leap to the conclusion that DTI is also using Craig Chamberlain's sector copier module without the author's permission. But one who leaps to conclusions frequently lands with his foot in his mouth and, inasmuch as I've been unable to locate Chamberlain for his comments, it's probably best to give DTI the benefit of the doubt.

Each of us probably has reached his own judgement with regard to the practice of "backing-up" commercial software. Personally, I see nothing wrong with it as long as the "backup" copy is used as just that — a "backup" copy to be used if the original develops terminal illness. "Back-ups" should not be given to your Aunt Tillie, the little kid who lives down at the corner, or your buddy who works in the next office. That's not just immoral, it's also illegal. And for the life of me, I can't think of a single reason why I might need a "backup" with the protection scheme deleted and access software codes removed so I can send the program to someone else via modem.)

From the Editor
by Rick Singbeil

If you've read Antic and Analog lately, you may have noticed the little feud going on between Happy and The Duplicator. It seems only one sided, as Duplicating Technologies Inc., who make The Duplicator, have not responded to Happy's claims.

This, however, is not the case. In talking to a friend from Thunder Bay about what he thought of The Duplicator, I was informed that it works fine. He also sent me a copy of a letter he received from Duplicating Tech., which all purchasers probably receive. Here's the letter which explains the other side of the story:

January, 1987

Dear Customer:
1. The 4.0 is ready.
2. About the Happy's Cheerup Upgrade.
3. The Real Story of Happy's Major Problems.

We are pleased to announce that Revision 4.0 for our Duplicator is ready and waiting to be shipped.

As we stated previously, our Module creator is enabling us to get our revisions out much more quickly and efficiently. We now have the technology to backup any form of protection on the market today. Even disks protected by hardware keys that must be present in the joystick ports offer no problems for the Duplicator.

The Module program will remove the protection from the copyguarded disk allowing our customers to make backup copies of their valuable software and these backups you make can now be sent over the phone lines by modem.

In order to demonstrate the power of the Duplicator and show our customers we now can backup anything, our 4.0 module has a great variety of copyguard schemes. We would like to mention a few of these now because we are very excited about them.

The 4.0 includes Racing Destruction set by Electronic Arts, World Karate Championship by Epyx, Lode Runners Rescue by Sunapse, Wizards Crown by SSI and Music Studio by Activision. Our main copy program now includes all the Hayden SAT Educational Series. Duplicating Technologies are the first to make a working copy of Paperclip by Batteries Included. The copies made do not need the hardware key present in the joystick port. We have pledged to our customers that whatever the software manufacturers DO, WE WILL UNDO.

The price for this upgrade will be $13.95 plus shipping and handling. It is unfortunate to have to charge for upgrades but there is a great deal of time and work involved in getting these upgrades to you. We will try our best to keep the cost down to a minimum. The price for the upgrade is 1/3 the cost of just one of the pro grams the module can copy. There are over 100 modules on this 4.0 Revision braking down to about $0.14 a piece for each program on the module.

THE CHEERUP UPGRADE

We understand one of our minor competitors the "HAPPY" is now selling a "Cheerup Upgrade" to turn your Dupliccator into a "Sad Happy". Please understand they are quite worried about the power of the Duplicator. (Read Part 3 for the real story. We have written a thorough explanation of this.) If you decide to Happy Enhance your Duplicator, please understand the copies of many programs you make will run only on that Happy enhanced drive. EXAMPLE: A copy of Zorro or Never Ending Story or even Goonies made on the duplicator will run on any drive. The copy made on the Happy Enhanced Duplicator will now run on that particular drive ONLY! You will not be able to run these copies on any other drive what-so-ever. Our duplicator copy program and software will not workk with this "Cheerup Upgrade".

We at Duplicating Technologies feel its a pitiful thing to ask a customer to buy a product (The Happy) and then have the software's execution dependent on that product.

We have our answer to the "Cheerup Upgrade". We call this product the "Backup Master". This product is not necessary for our Duplicator customers. Our 4.0 contains a variation of the "Backup Master" combining the long awaited Telegard program with the Duplicator. You now can send all your module made copies over the modem lines. Our "Backup Master" and 4.0 modules will actually remove the protection from your copies.

The "Backup Master" is software only. It has been designed to backup hundreds of individually listed programs without the use of our Duplicator. It has also been designed to help the Happy drives copy disks that:
1. cannot be copies with the Happy and
2. can be copied with a Happy but now can run on any drive.

We do not ask you to change your Happy. The Backup Master does not make any changes to your Happy. What we do is allow you to break your chains to the Happy drive and let the copies you make run on any drive. The Backup Master does not even NEED a 1050. You may use your RANA, INDUS, TRAK, or PERCOM to make your copies. Look for more information about this product in our upcoming advertisements in Antic Magazine or write us for a detailed explanation of this product.

OUR OPINION OF HAPPY COMPUTERS

The release of our REV 3.0 software for the 1050 Duplicator has greatly angered one of our minor competitors Happy Computers Inc. of Morgan Hill California.

To those Atari users who are not familiar with Happy Computers, they are a small outfit run out of Richard Adams home in California. They market the Happy 1050 Enhancement for the 1050 drive. This device is based on the 6502 microprocessor and is similar to our 1050 Duplicator. This device has merit. It will increase the disk loading time and give your drive true double density. The Happy Enhancement does have one major weakness, which is it CAN NOT MAKE WORKING BACKUPS OF THE NEWER FORMS OF PROTECTION APPEARING ON THE MARKET TODAY. Its inability to backup new programs we feel will eventually put Happy Computers out of business. We will explain further on.

The reason for writing this letter to Atari users is Duplicating Technologies has been the victim of some rather unflattering advertisements by Happy. We feel it is time to reply to his questionable business practices. Although we have never used his company name in our advertisements and we have never sent customers poor reviews of his Happy written by us or disinterested third parties. It is time to tell our side of the story.

The bottom line is that the "Happy" has major major poblems.

Software manufacturers can use the MOST SIMPLEST copyguards like the bad sector to stop piracy and the backing up of their software.... Then why do they use such elaborate and sophisticated forms of protection? The answer is simple to make products like the Duplicator and the Happy as ineffective as possible.

A new form of protection has appeared on the market recently and it is called "THE WEAK SECTOR" (we will go into the technical aspects of it a little further on.) This new form of protection is being employed by companies like Datasoft and Synapse at the moment. What Synapse and Datasoft has done is make a protection scheme so powerful; it knocked the Happy for a "major loop". It made the Happy unable to make a working copy of the program run on any drive except its own drive. Can you imagine the consequences of this protection to a Happy Enhanced drive. If other software manufacturers decide to employ it and the weak sector becomes more and more widely used, all these new programs will need a Happy drive to run its copies. "Give your original to your one friend and keep your copy for your own drive". Don't you think Synapse and Datasoft are trying to stop the spread of their software. They realize they have done it with Happy. You can no longer make those indiscriminate copies with your Happy because they will not run on any of your drives except the Happy Enhanced drive.

Richard Adams will shyly tell you this is true but, the Duplicator cannot copy everything the Happy can. This maybe true although only Syncalc comes to mind that can be copied with a Happy and not a Duplicator. Syncalc will be available on our 5.0 upgrade. If you find anything the duplicator cannot copy with this 4.0 Revision write us and we will have it taken care of with the following upgrade. The point is we can copy the weak sector. There is no copyguard scheme on the market today we cannot copy and have it run on any drive!! We have demonstrated we can do all the protections schemes on the market today.

Now lets give you examples of what we are talking about and let us get a bit technical.

WHAT IS A WEAK SECTOR?

A weak sector must be created on a very sophisticated and costly commercial disk backup device used by large software protection houses. When this weak sector is written; the current to the read/write head of the drive is lowered for a few micro seconds. The result is a sector that is called "weak". Everytime you read this weak sector, the data will be slightly different where as a good sector will always return the same data. Once a weak sector is on a disk it is easy for the programmer to protect his software for he knows every time he instructs his program to read this weak sector it will return different data.

Let us say as an example we wanted to copy protect a game and we knew that sector 719 was weak we would write a small routine to read sector 719 say 5 times and if the data of the 5 reads were to be different each time we would then branch and run the game. If the reads of sector 719 were the same each time then we would assume the disk to be a pirated copy (for standard hardware cannot reproduce the weak sector) and we would have our program branch on the condition and lock up. When the Happy drive reads a weak sector it will write a good sector in its place. It is absolutely impossib e for the Happy to reproduce a weak sector. His hardware does not have this ability and never will. As stated previously the reason for the costly development of new forms of protection by software manufacturers such as weak sectors is to put copy devices like the Happy Enhancement and the 1050 Duplicator out of business. A simple bad sector is all that is needed to prevent copying software with DOS or copy programs that are sold. Only a few games are now protected by the weak sector for this protection is new, but one thing is for sure it will spread and the day will soon be here when all software manufacturers will employ the weak sector as a copyguard. Why? Because it works! In time this will render the Happy 1050 useless. Happy will be history for he will be out of business. Duplicating Technologies has demonstrated the ability to copy the weak sector. Our future and that of our customers is secure.

Happy claims to copy the weak sector but this is trickery. Let us use the game Zorro by Datasoft as an example. It is a good representation of all weak sector programs. THE COPY MADE BY HAPPY WILL ONLY RUN ON A HAPPY DRIVE. This is absolutely and TOTALLY UNACCEPTABLE. What if your Happy fails? You will not be able to run your backup. What if you have more than one computer system? You will be forced to buy a Happy for each drive. What will you do if a friend or neighbor wants a copy? In his latest brochure he says "his way will stop the spread of software piracy for the copy you make will only run on a Happy". Come on now!!! DOES HE THINK ATARI USERS ARE STUPID? All other copies on the

(13)

Happy will run on any drive. ALL OF A SUDDEN HE IS CONCERNED ABOUT PIRACY. Whose side is he on? The users or the software manufacturers? Why doesn't he admit the truth? He cannot do what the 1050 Duplicator can do. The staff at Duplicating Technologies and Bob Gardner have put him to shame! Richard Ames and his Happy Enhancement have some problems.

The copy you make of the game Zorro made on a 1050 Duplicator with Rev 3.0 or higher software will run on any drive and any Atari computer. There are no restrictions on our copies. Another important fact is that we do not force Atari users to buy our products in order to run backups and we never will. Happy Computers is no longer competition to Duplicating Technologies. His inability to make all copies run on any drive removes him from this category. We would like to compare the Happy to the Impossible from Computer Software Services. Their copies only run on their computers.

Duplicating Technologies will not be put out of business by new forms of protection and our customers will always be able to make working copies (on any drive) of these new programs. We would like to say one more thing to clear the air, that is Richard Adams has been telling people we copied his idea. I did not know ideas were private property. RCA had an idea for a television, I do not seey the Sony or Zenith television infringing on their idea. The same applies to G.E.'s refrigerator infringing on Whirlpool's refrigerator.

I will tell you a fact. The Happy copied the entire Atari 1050 operating system and put it in their own operating system which is the cause of their problems today. Our operating system was written in machine language from "byte one" byte for byte. This is why our product needed a shakedown period this past year. It is stronger and more powerful than any Atari operating system and any Happy and its future is tremendous.

We are in the process of designing new and innovative software to further the duplicator's efficiency. Our compiler is nearing completion and our modules are even better. Our products are doing very well in Canada, Europe and of course the United States. We will support our system at all costs and go to any lengths.

The Staff at Duplicating Technologies

* * * * *

** For Sale **

Atari XM301 serial modem (like new)
Manual & Terminal program
$ 50.00

Atari XMM801 serial printer (like new)
Manual & Graphics driver
$ 285.00

call Stu Marcellus - 6682178

* * *

Hayes 300 baud SmartModem AA/AD
Manual & Cable to 850 Interface
$ 150.00

call Rick Singbeil - 8890098

* * * * *

NEXT MEETING

M.A.C.C.
Grant Park Harrow Recreation Association
550 Harrow Street
First Thursday of the Month
March 5 1987
7:30 P.M.

* * * * *

M.A.C.C. BBS
300/1200 BAUD
phone 257-0535

* * * * *

(MICHAEL, cont.)

Teacher: Take the green wafer marked mathematics from the kit in front of you. Insert the flat side first into the recepticle in your belt. Now class, what is the cosine of 154.78739?

Class (in unison): .63998632733

SCENE IV

Luther's home.

House: So happy to have you back, Luther. I hope your stay at the hospital was not too traumatic.

Luther: No, it was OK.

House: I have a message from your parents. (The room darkens and the images of the parents appear.)

Father: Sorry we couldn't be there when you got home, but we had some extra work to do.

Mother: Yes, it was unavoidable, but we didn't forget your present. We told House what to buy, and she has it for you.

Father: We know how much you like to draw, so we got an art wafer for you. I hope you like it. We will be home as soon as we can. Bye for now.

Mother: Bye Luther.

The mother and father disappear and the room lightens. Luther walks to the table and finds a wafer there. He examines it and inserts it into his belt. He looks about the room with a new view. He sees the Crayon drawing on the table and studies it. He takes a clean sheet of paper and quickly draws another picture of his mother, but this one is perfect in the style of Michelangelo. He compares the Two, smiles and puts the Crayon drawing into the disposer.

The end

THE EXPANDER OS (14)
ACCESS KEY 2-87

Ray Tonkin has developed a new hardware device for the XL computer which consists primarily of a new Operating System on ROM together with 512K of expanded memory. He calls his product "The EXPANDER Operating System", and he will be demonstrating it at our February meeting. Some of The EXPANDER'S capabilities are:

* ROM resident.
* 98% compatible with 800 software.
* Basic in/out through RESET.
* Menu driven monitor program.
* 9 options of Ramdisk control.
* Boot any one of 4 drives at anytime.
* Make any Ramdrive boot as Drive #1.
* Boot programs at Ramdrive speed.
* Break any running program & reboot.
* Change drives from DOS.

In addition, his system comes with G-DOS, which will format your disk - RAM or floppy - and transfer data to and from RAM and floppy. Thus, you can use the Ramdisk for fast and important data transfer, and then copy the contents onto floppy for permanent storage. You can also put your favorite program in Ramdisk, boot it up at high speed, and when you're finished with that program hit RESET and reboot the Ramdrive.

Ray plans to market The EXPANDER through user groups, at least to start with. So, I expect he will be ready to take your orders at the meeting. Sorry, I don't have a price to quote you.

FUTURE CLUB MEETINGS

APES SIG 5 March
   (Call Francine at 967-9271)
ST SIG 9 March
   (Call Dave at 773-4548)
PROGRAMMING SIG: 18 March
   (Call Bob at 421-1573)
TELECOMMUNICATIONS SIG: 3 March
   (Call Barry at 454-3113)
NEWSLETTER DEADLINE: 27 February

GENERAL MEETINGS: 25 February, 25 March
   22 April, 27 May

Inside the Black Box

by rootbeers

This month's episode:
*Basic Hardware Concepts*

HACKS
2-87

This month I'd like to introduce some of the fundamentals of hardware found in the Atari 8 bit computer. I have recently become intrigued by the hardware side of computing and hopefully this will stir some interest in the field. By far the hardest part is learning the terms used with hardware—everything else is similar to programming.

For the most part, this discussion will deal with the DIGITAL field of electronics. Digital electronics is primarily concerned with two voltages, affectionately called *LOW* and *HIGH*. A wire can be considered to be at either a low or high voltage. The values 0 and 1 are often substituted for LOW and HIGH to aid in writing and computation. TRUE and FALSE are also used to refer to the HIGH and LOW states (0 = LOW = FALSE; 1 = HIGH = TRUE, in general). A STATE is a specific assignment of values to a set of wires. (These are really the same con-

FIGURE 1

DIP PIN NUMBERING

THE TOP OF THE CHIP IS MARKED BY EITHER A PRINTED STRIPE, A DIMPLE ON THE LEFT TOP COR-NER, OR A NOTCH IN THE CEN-TER OF THE TOP EDGE.
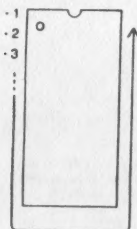
cepts as binary numbers. A wire is about the same as a bit; a state is about the same as a binary number.) If a digital circuit produces output states solely determined by its input state, then the digital circuit is said to be a COMBINATION circuit, and for now our discussion will be limited to them. In the Atari computer, the voltage level of LOW is zero, or GROUND; The voltage level of HIGH is +5 volts.

In digital electronics, there are GATEs which have one or more INPUT wires and produce one or more OUTPUT states based on the state of the inputs. A gate performs some function on the input values to obtain the output value and the function performed is the type of gate.

FIGURE 2

JOYSTICK PORT PINOUTS
(PORT 1, VIEWING PINS OF THE COMPUTER)

```
  1 2 3 4 5
   6 7 8 9
```
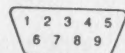
1	BIT 0 OF PORTA ($D300; 54016)
2	BIT 1 OF PORTA
3	BIT 2 OF PORTA
4	BIT 3 OF PORTA
5	PADDLE 1
6	TRIGGER
7	+5
8	GROUND
9	PADDLE 0

For example, a two input AND GATE has an output of 1 if both the input values are 1, and a 0 otherwise. AND, OR, and NOT gates (NOT gates are usually called INVERTERS) are available. As luck would have it though, the most popular types of gates are NAND (not-and) and NOR (not-or) and they operate exactly like an AND followed by a NOT and an OR followed by a NOT, respectively. So, the output of an inverter is HIGH only if its input is LOW. The output of an AND gate is HIGH only if all of its inputs are HIGH. The output of a NAND gate is HIGH if any one of its input lines is LOW. The output of an OR gate is HIGH if any of its inputs is HIGH. The output of a NOR gate is HIGH if all of its inputs are LOW.

So, we're concerned with circuits where any place on the circuit can be said to be in a high or low state for all possible input states. The

circuits consist of gates which do logic based on the inputs of the gate and produce a resultant output value. The output of one gate can of course be used for another gate's input.

Gates are packaged in 14- or 16- pin DIPs (*Dual Inline Package*: the little black chips featuring two rows of connections that look like little electronic insects). Chips with under about ten gates on them are referred to as SSI (*Small Scale Integration*) chips. There are many ways that the actual gate may be implemented, and the implementation you find used most often on your Atari are TTL (sometimes T2L; *Transister-Transister Logic*). TTL SSI chips have code numbers which start with the digits 74. The pins of a DIP are numbered as in Figure 1. Usually, the highest numbered pin is to be set to a +5 source voltage and this pin is often labeled in the manuals as *Vcc*, and the pin diagonally opposite the Vcc pin is sometimes labeled *Vss* and that pin is set to ground. Together, Vcc and Vss provide power for the chip.

So that is a brief introduction into the alphabet soup that you'll see if you try to look at a specification sheet for a chip. Let's take a look at some of the things you don't want to do: 1) Don't connect two output pins to each other even if they are both at the same state. At worst, this would destroy one of the outputs of the chip. 2) Never connect the +5 voltage directly to ground; you could blow your power

FIGURE 3

SCHEMATIC REPRESENTATIONS

OUTPUT →

INPUT →

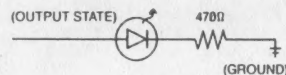AND   NAND   OR   NOR   XOR   INVERTER
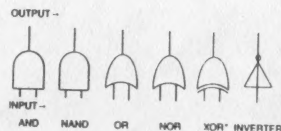
supply. 3) If you use the output of a gate, be sure all of the inputs of the gate are connected to something; an unconnected input to a gate is said to be *FLOATING* and its state is not explicitly determinable, so this is considered

poor practice, but won't result in any damage to the circuit. If an input is connected directly to a source voltage, it is often referred to as being *CLAMPED* or *TIED* to that state. TIED is also used to imply any electrical connection.

I'm sure it's obvious that these fundamental concepts are the building blocks used to build a microcomputer, but it's probably less obvious that the basis for beginning experimenting with hardware is as near as your Atari! In Figure 2 you'll notice that the joystick ports allow access to +5 and ground voltages, and while I don't recommend doing so, I do know that shorting the +5 to ground (the no-no mentioned above) does not destroy the Atari power supply. In addition, four of the pins on the joystick port can be set to either output or input values from your circuit, so that a program you write could control or monitor the circuit you build. Com-

municating with a circuit via the joystick ports is an article in itself, so we'll save that for a later date.

To start experimenting you would, at the most, need to invest less than $20 to buy: some "perf board" (circuit board material with no metal (TRACES) on it, just a grid of holes), a wire wrap tool and wire, some LED's (*Light Emitting Diode*), some resistors, some wire wrapping sockets, a nine pin D-connector female (like the end of a joystick cable), and a few SSI chips. Everything but the SSI chips should be available at your local Radio Shack. Personally, I'd pick up the SSI chips at Priority One, or a similar electronics store regardless of whether Radio Shack stocks them.

I've included a few drawings to show how the

FIGURE 4

USING AN LED TO DISPLAY AN OUTPUT STATE

(OUTPUT STATE)   470Ω

(GROUND)

LED'S SHOULD BE INSTALLED SO THAT THE LOWER LEAD (THE "CATHODE") IS PULLED TO GROUND. THE RESISTOR IS NEC-ESSARY TO KEEP THE LED FROM BURNING OUT.

various gates are shown on a schematic (Figure 3), how an LED is connected to show an output state (Figure 4), and how a switch can be installed so that the output from the switch will be guaranteed to be in either a low or high state (Figure 5). Note that when the switch is open, the output line is connected to one of the power terminals (in this case ground) through a resistor. This is called *PULLING* and in this case the output is pulled low. When the switch is closed, the line is at a high state and the resistor keeps the +5 source from being connected directly to ground. Pulling is fairly similar to setting a default state.

Well, I'm sure that that's about as much as you can learn on the subject in one article, but there are some final things to be said about hardware experimenting in general. The terminology of digital electronics differs from software terminology, but the concepts are really similar. It is possible to learn and to build gadgets with only a little knowledge. The Atari 800 seems fairly robust as a workstation (by which I mean that I have not yet succeeded in destroying mine, in spite of numerous accidents), so I think the risk to life and machine from experimenting is pretty small.

For those of you whose interest is caught by this subject, please say hello either in person or via the HACKS mailing address. I'll be happy to answer any questions or listen to any sugges-

FIGURE 5

A "LOGIC SWITCH"

+5   OUTPUT STATE

470Ω

(GROUND)
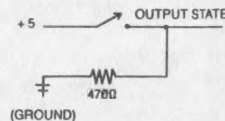
tions. If you do decide to dabble in digital electronics, I suggest you get your hands on the Atari 800 schematics; some of the gates you will be able to recognize just from the information in this article, and as you learn more they

will begin to make more sense. The book "Digital Logic and Computer Design" by M. Morris Mano is a good text for learning about how to design digital circuits, and any reference book which shows the PINOUTS (specifies the use of each pin of the chip) of several SSI chips comes in real handy. Most of the reference books I have at home are at least two and sometimes ten years old, so don't worry a lot about whether the book is real current. Remember that the technology of tomorrow costs alot more than the technology of yesterday. Finally, the bookstore of Cal State Northridge sells a laboratory manual for about six dollars which would be a good initial set of projects to build ("Digital Circuits Laboratory Manual" by Nagi M. El Naga, Course = EE355L). The laboratory manual contains a variety of projects and information, including some simple schematic diagrams and pinouts of all the chips used.

INKFINITY FOLLOW-UP
WAND 2-17
by Rolly Herman

Every once in a while, a product comes along that is the answer to a maiden's prayers. Such a product is Inkfinity. I wrote a review of it in the September issue of The W.A.N.D. and promised a follow up. Here it is. (See the Inkfinity "ad" in the Sept. '86 issue.)

Since September 1986 I have used Inkfinity exclusively for my Juki 6000 Daisy Wheel printer and for my Epson MX80. On the Epson, I have used it for the color ribbons, as well as the black. All work to perfection.

This is a system that is almost too good to be true. In a computer environment of complexity, Inkfinity defies Hi-Tech with its simplicity. A small piece of felt and a bottle of ink comprise the entire system. But more amazing than the simplicity is the result. Many things that are simple give compromising or mediocre results. Not so with Inkfinity. I have better results with my printouts, using Inkfinity, both in color and black, than with brand new ribbons. And the cost is also unbelievable. The kit is only $ 5.00. I don't know how long a bottle of ink will last. In the past six months I have used less than a quarter of the bottle. I used to buy new ribbons every two to three months. Now I use the same ribbon and just add a few drops of ink to the felt when the printout begins to get light. The ribbon does not have to be removed and the procedure takes about 10 seconds. No fuss, no mess. I assume that the ribbon itself will last a very long time. Eventually, I suppose, the pins in the printhead will actually wear through the fabric of the ribbon, but until that time arrives I do not have to buy a new ribbon.

Color ribbons were always a problem because they have a tendency to dry out when not in use. Inkfinity solved that problem completely. When I started using the YEMACYB program for color printouts, it was necessary to use red, blue, yellow, and black ribbons. Inkfinity does not sell yellow ink (perhaps they will when they read this article), so I ordered yellow ink from Computer Friends, 6415 SW Canyon Ct. Suite 10, Portland, OR 97221, (503) 297-2321, and use it with the Inkfinity felt. It works perfectly. (Computer Friends makes a machine called the MacInker for re-inking ribbons.) Incidentally, the YEMACYB is a fabulous color printout program reviewed in The W.A.N.D., Nov. 1986.

Wordprocessing is one of the main uses of home computers, and the Atari machines have great graphic capabilities. Therefore, our printers must be getting lots of use preparing documents, graphs, page layouts, digitized pictures, color prints, desktop publishing, etc. If your ribbons are getting light all the time and you are frustrated by not having spares, or by being obliged to buy and change ribbons often, then it is time for you to switch to the Inkfinity system. I will be glad to help anyone glue on the felt pad and get started, or to answer any questions. Give me a call. 914-946-4134.◇

DIANE'S DATA
by Diane Bittegar

AACE welcomes the following members to the user's group:

David Bukovsky
Kenneth Collings
J.C. Foltzer
Scott Guthry
David Ishem
Steven Swaim

Charles Clemer
Paul Durry
Kieran Garvey
Rudolph Green
Ben Hobbs
Mike Jankowski
Willie Miles
James Timberlot
Debbie Timberlot

The following members have renewed

Allan Adams
E.E. Baden
Quinn Credi

Edwin Arnold
Bernie Bennet
Mike Jackson
Bob Myers

Kyan Strikes Back!
by A Reasonably Informed Source

Chapter One
WHAT HAS GONE BEFORE:
(version 1.1)

A long, long time ago, in a galaxy far, far away, KYAN unleashed an Atari (8-bit) version of KYAN PASCAL upon an unsuspecting universe, which soon became the most powerful Atari 8-bit PASCAL programming language in the cosmos.

This PASCAL did have two weaknesses, however, which allowed the rebel programmers to attack it.

This is the story of these weaknesses, and the steps that the KYAN empire took to counter them...

* VERSION 1.1 *

The two major problems with KYAN PASCAL V1.1 were: the poor manual and the lack of DOS accessibility, both within the programming environment and from the program itself.

These two problems made a fine PASCAL programming environment very difficult to work with. Let's take a look at what the problems actually were.

* THE MANUAL *

As I have stated many times before, the KYAN PASCAL manual was the WORST manual that I have ever worked with. It was basically the manual for the Apple computer version with brief notes pertaining to the Atari version. These notes were often unclear and sometimes inconsistent. Typos in strategically located positions, downright incorrect information, and lack of actual implementation examples made it difficult to learn how to edit and compile, not to mention learn new programming methods.

The resulting trial and error programming in a compiler-based language does not make for a very steep learning curve.

When programming examples were given, they were usually either too simple or too complex to be of practical use.

At one point, the Atari instructions were so bad, I actually had to refer to the Apple section and play it by ear!

All in all, the manual made the learning and implementation of PASCAL a real chore. If it were not for the excellent SAM'S PASCAL PRIMER (see RIACE REPORTER #26, "Kyan Pascal am's Pascal Primer"), I don't think I would have gotten very far with PASCAL at all.

The KYAN PASCAL manual was approximately 108 pages; PASCAL is a tough language to fit into 108 pages. Also included were a few Atari-specific addendum pages. This was all bound in a clear plastic binder of acceptable quality.
* DOS ACCESSIBILITY ILE MANIPULATION *

FRANCISCO'S
ROADSIDE SOFTWARE

SE HABLA
PASCAL

chase

(continued from page 7 Kyan)
PASCAL was two-fold. First, the non-accessibility of the DOS commands from within the programming environment. To format a disk, lock or unlock a file, move a file to the ramdisk, or read a disk directory, you had to go to DOS, do what you had to do, and then BINARY LOAD (L) the PASCAL environment again. As often as I access the directories, this was a pain in the you-know-what!

The other difficulty arose from the fact that there was no way to access directories from within an executing program. As my programs tend to do a lot of text file manipulation, combined with PASCAL's lack of error trapping, this makes much of my programming unstable.

KYAN does offer a systems utilities disk that offered such things as disk directories, but as the price was approximately $50.00, which is what I spent for KYAN PASCAL in the first place, I couldn't bring myself to lay out that much money for it (especially with rumors of an upcoming update that would take care of many of the forenamed problems).

* ODDS NDS *

In addition to the poor manual and "DOS-lesness", there were a few other problems that hindered programming with KYAN PASCAL, such as:

1. KYAN PASCAL would only run on an 800XL or 130XE. If you had an unmodified 800, then you were out of luck (although an upcoming 48K version was promised).

2. Identifiers could not include an underscore character. The variable RATE_OF_PAY is

much more readable than RATEOFPAY.

3. KYAN PASCAL booted from a modified DOS 2.5, therefore you couldn't use a more powerful DOS, or use true double-density storage.

4. PEEKs and POKEs were excluded.

5. The GOTOXY (cursor positioning) procedure was left out. This made screen formatting not fun.

With the exception of the problems and shortcomings mentioned, KYAN PASCAL was an exceptional value for $69.95 (less through discounters). A few other PASCALs were available for the 8-bit Atari, but KYAN PASCAL was (and still is) the Imperial Battlestar of PASCAL compilers for the 8-bitters.

Chapter Two
KYAN PASCAL STRIKES BACK

A while ago (quite a while - this article missed deadlines), I received notice that KYAN PASCAL was now updated to Version 2.01A, and for $24.95 ($20 for the update and $4.95 shipping), and the return of the original V.1.1 disk, I could receive my update package.

Although at first I thought that $24.95 was quite high for an update, I sent my money and disk. A couple of weeks later, my trepidation was eased somewhat when I received a new, complete KYAN PASCAL package. Instead of copying the update onto the old disk, or just sending a new disk, they sent a whole, brand new package, just as if I had ordered a new one.

Let's take a look at what KYAN did (and didn't do) to improve their PASCA

* THE MANUAL II *

The new manual is a definit[e] improvement over the old, i[n] many ways!

First of all, it is now 30[0] pages and is bound in a[n] attractive, high-quality three rin[g] binder, and is no[w] ATARI-SPECIFIC! The editing and compiling instructions are much better written.

Each chapter begins with an OVERVIEW which introduce[s] the upcoming concepts. Following is THE PROGRAM which is built upon the relate[d] concepts. THE LOGIC OF THE PROGRAM is a step by ste[p] explanation of the program. Then comes a few GENERAL COMMENTS followed by TOPICS FOR PROGRAMMERS and ADVANCED TOPICS (o[r] TECHNIQUES), which introduce[s] the more refined of PASCAL commands. Last is the CONCLUSION which sums up the previous chapter and introduces the next one.

The example programs supplied are much more suitable for learning, and do not get too complex too soon.

Although the concepts of PASCAL programming are better explained, SAM'S PASCAL PRIMER, or another good PASCAL book is still a must.

The manual's coverage of including ASSEMBLY LANGUAGE within you[r] PASCAL source code is greatly expanded, although it is not meant as a tutorial. A good deal of A.L. programming experience is required. NOTE: a few changes in A.L. code must be made if you wish to re-compile

code from V.1.1 on V.2.01A, although the changes are minor.

The list of error messages has been expanded from 35 to 141 reportable errors.

All in all, the new manual is well written and well laid out. The quality is first rate. It almost justifies the high update price. The amazing thing is that the $69.95 retail price was not affected.

° NEW FEATURES/CHANGES

KYAN PASCAL now includes KIX, which allows you access to DOS commands from the PASCAL programming environment, but unfortunately not from the program itself (looks like I'll need to buy the utility disk after all).

KIX is actually a series of files which can be called from the system prompt (which has been changed to % as opposed to the older >).

Following are the KIX commands and what they do:

PWD - (Print Working Device) Prints the number of the current default device.

CD - (Change Device) Changes default drive.

LS - (List Directory) Prints directory to screen, printer, or a file.

CAT - (Concatenate) Appends one or more files to a new file.

CP - (Copy) Copies files - use to move files to Ramdisk.

MV - (Move) Renames files.

RM - (Remove) Deletes files.

CHMOD - (Change protection Mode) Locks or unlocks files.

FORMAT - (Format disk) Formats a disk in either single or enhanced density.

SD - (Screen Dump) Copies the screen contents to the printer.

While working with KIX, you need not specify device numbers. KIX will automatically search first the Ramdisk, then the user device, and then the default. KIX will also accept both upper and lower case, converting all to upper case.

The only thing lacking is the ability to batch process KIX commands. It would make setting up your Ramdisk much easier, for instance.

° ODDS NDS II °

A few of the minor complaints have been cleared up in Version 2.01A.

1. KYAN PASCAL V.2.01A will now run on any 48K Atari.

2. Underscores are now accepted.

3. It will now boot from most DOSes.

4. Two ASSEMBLY LANGUAGE routines are included to allow PEEKs and POKEs.

5. Unfortunately, GOTOXY is still NOT supported, but the manual explains how to use the POSITION graphics procedure to position the cursor.

Kyan Pascal will be contined next month so stay tuned right here on the RI ACE Reporter for Kyan Strikes Back Part II by a Reasonably Informed source. Ed.

Austin Atari Computer Enthusiasts--AACE
January 1987

told electric devices to move the print hammers. Sometime after that, someone had the idea to install a port—to get at the signals at a point between the keys and the print mechanism. You could then do different things than just drive the print mechanism. For example, you could send the signals over long distance to drive a print mechanism elsewhere, or store the signals in some way for later retrieval. You could even manipulate the signals in ways different from what you could do when the signals had deposited ink on a piece of paper—word processing.

In the beginning there was the piano (and harpsichord, organ, etc.). The keys were attached directly to the hammers. Then came the electronic synthesizer, where the keys were electronic switches that controlled electronic sound generators. A number of inventors (Robert Moog, Don Buchla, and others) realized that a standardized signal path would provide much greater potential for technical growth. This standardized (well, sort of) signal path was the control voltage. Instead of turning a knob on a sound generator, you could supply it with a control voltage that would do the job for you. You could route these control voltages in many ways, using patch cords (cables). You could even store the control voltages in some form and play them back later.

With the development of less-costly computers, there came a need for a standard form of interchange of data. The conversion of information to several evolving forms. There was the five-bit Teletype code (remember those old style Teletypes)? LEAVING PITTSBURG SATURDAY STOP ARRIVING DENVER THURSDAY STOP ARE YOU HAVING TOO MUCH FUN QUESTION MARK IF SO STOP. That was a result of a five-bit code, which allows only 32 characters—they didn't have enough characters for punctuation.), the eight-bit EBCDIC (Extended Binary Coded Decimal Interchange Code) used by IBM mainframes, the six-bit Display Code used by CDC, and ASCII, which is recognized by almost everyone nowadays.

As computer power got ever more affordable and available, the synthesizer manufacturers recognized the need for a standard digital interface. (Control voltages are analog—that is, they can take an infinite number of values within a given range. Digital information can represent only exact values, an 8-bit digital number can represent any whole number from 0 to 255, but cannot represent 17.37 or 3.333 or 3.1415926535897931. This makes analog signals sound more powerful, but digital signals are much easier to design complicated circuits around. In 1982, several synthesizer manufacturers led by Dave Smith of Sequential Circuits, got together and agreed on a standard form of musical information interchange --MIDI.

ASCII is a standard code—well, sort of. If you send an ASCII the number 65 (hex 41), the printer will almost always print a capital letter "A". What happens if you send, say, 193? Some printers only recognize seven bits, and will map this number back to our friend "A." Others think ASCII in an eight-bit code, and will give you something different. An underscore character, say, or reverse video, or graphics character. And how are you sending this digital information? Through a parallel port, maybe? What is your handshake protocol? Or maybe you're using a serial port. Are you using the RS-232 "standard?" (Ha. Ha ha. Ha ha ha ha ha...) What's the baud rate? How many stop bits? Which pins on the 25-pin sub-D connector are you using?

The guys who came up with the MIDI standard had been through a lot of this

---

AACE--Austin Atari Computer Enthusiasts
January 1987

complete desktop publishing system, but still want graphics, text and multiple fonts, not to mention a VERY powerful word processor, MS Write will be for you! The version I saw was 95% complete, and virtually bug free...so I suspect most of the delay is probably in packaging, and the issue of $$$ between Atari and Microsoft You can bet that Microsoft will include lots of printer drivers. Even there IBM version has something like 250 different printer drivers! The expected price I have heard will be $149.00 retail.

Moving on to the 8-bit, I got a few letters from some 8-bitters that placed interest in there investment, by pleading for more 8-bit coverage through-out the year. Well, my only response is that I will try to put as much 8-bit news in this newsletter as I can. Alot of users groups across the nation have discovered the same problem as ours...there just isn't too much happening with the 8-bit. But, I will try!

I heard that either some third party group, or the Atari 8-bit Developers themselves, were developing some new OS Routines, that would allow for a Graphics 8 resolution, 4-color model Sounds pretty interesting!

Have you seen Alternate Realities on the ST yet? This 8-bit conversion was well worth the wait, it features great graphics and sound, and in on a one disk system (no more major disk-swaps, like the 8-bit version). Also, Alice Pascal is really a slick interpretive Pascal for the ST. It has step by step debugging, and uses a syntax directed editor. What this means, is that when you type the standard line:

For X := 1 to 100 DO,

in this editor, you would just hit FOR, followed by a <TAB> and it would fill in the <= and put in different colors the words START TO FINISH DO, so all you do is fill in the blanks...essentially not allowing you to make any syntax errors! (What a blessing). The package sells for $79.95 retail.

The Aerco 1-Meg Boards are really nice. I just installed mine last night, and it took less than 10 minutes—no soldering, and it was less than an arm and a leg. $189.00. Best of all, the boards are made here in Austin, so there is local help available for any problems that could pop up.

If you have any article submissions, don't hesitate to upload them to the Software Exchange BBS--ph# 448-4282.

Well, until next month...¶

The Well Tempered Atari
by Ed Comerford

My first column was about me. My second column was about the profusion of MIDI software just arriving on the market. This third column is about MIDI, a term I have so far used without explanation. What Is This Thing Called MIDI?

"MIDI" is an acronym for the Musical Instrument Digital Interface. Well, that gives you as much information as saying that "ASCII" is the American Standard Code for Information Interchange. Let's compare MIDI to ASCII for a bit here.

In the beginning there was the typewriter. The keys were attached directly to the print hammers. Then there were electric typewriters, where the keys were electrical switches that

The following line appeared in the May, 1979 issue of Byte Magazine, page 141:

"...Memory for peanuts... 32K for $620, 16K for $315"

And you thought $200 was alot for a 1-Meg upgrade!

sort of thing. They standardized the numerical codes transmitted and received by synthesizers, and also standardized the fine details of the electrical interface to be used for these codes. A MIDI port is a five-pin DIN jack. The MIDI signal travels in a current loop using pins four and five. Pin two is connected to the shield of the MIDI cable. Pins one and three are not connected. (The MIDI Out port of the Atari ST is not a standard MIDI Out, because the MIDI Thru signal is traveling on pins one and three).

The Baud rate is 19,200 (yes, 12 times as fast as a good modem--but it's not fast enough for some people. Details in a later column). The MIDI cable must be a low-capacitance, two-conductor plus shield, maximum length fifty feet. (The new Abacus book, Atari ST Introduction To MIDI Programming, contains a little passage wherein the authors express their dismay over the cost of MIDI cables, and explain that Radio Shack sells MIDI cables for much less. Wrong. Radio Shack sells a cable with 5-pin DIN plugs on the end--but all five pins are connected. This might work fine. Personally, after I've spent $2.5 kilobucks on equipment, I am not very interested in saving ten dollars or so by using substandard cables).

The MIDI numerical codes are not as completely standardized as the interface electronics. This was deliberate--when the MIDI standard was defined, no one was sure what information would be good to include. If you connect a MIDI cable from the MIDI Out of synthesizer A to the MIDI In of synthesizer B and press a C# on A, synth B will play C#--but it might play it in a different octave. If you change programs (select a different sound for the notes) on synth A, B will probably change too--but unless the synthesizers are from the same company, the new sound of B will probably not sound like the new sound of A. (For example, when I select Preset 4 on my Akai AX-80, it sounds like a

harsichord--but the Akai sends Program Change message 4 to my Casio CZ-101, which selects Preset 4 there, which is String Ensemble 1). The most notorious example is the mismatch between the two most popular synthesizers, the Yamaha DX7 and the Casio CZ series. The Data Entry slider on the DX7 can be used to alter the DX7 voice while playing. But when the slider is moved, it sends a command that the Casio thinks is a command to tune the voice--in other words, using the slider on the DX7 makes a Casio CZ-101 (or CZ-1000, 3000, or 5000) go out of tune. There are ways to deal with these problems, which might become material for a later column. Newer MIDI instruments use a facility called System Exclusive for passing information that is not yet defined in the MIDI standard. This is a sort of agreement to disagree. When one instrument send a system-exclusive command to another instrument of the same brand, instruments made by other manufacturers ignore the stream of data until the transmitting device sends an End-Of-System-Exclusive byte.

The development of the MIDI standard in some ways reminds me of the origin of the Qu'ran (which we unwashed infidels usually call the Koran). Some time after Mohammed died, a caliph began a research project to compile the holy book of his words. The scholars labored long, and finally came to some agreement of what the Prophet had actually said. They presented their work to the caliph, who approved it--and then ordered that any documents that did not agree with this official version were to be destroyed. No messy doctrinal disputes--no Shah James Version or Good News Qu'ran--just the Qu'ran. Allah Akbar!

Next month--a comparision test of MIDI sequencers.¶

How can you describe to a nonbeliever the joy of working on a program till 2:30 in the morning?

ST Karate
by Paradox/Eldersoft
review by Brian J. Wilson

As you may have guessed by the title, ST Karate is a karate simulation which, in this instance, manifests itself as an arcade style action game. I refer to it first as a simulation primarily because of the high level of graphic detail and realism of the first software I've gotten for my ST that I don't generally run with the sound turned down.

ST Karate requires that you have an Atari ST system with a color monitor and at least 350K of free ram. You also have to have a joystick in order to play (and given the complexity of the controls, I suggest you acquire as good a joystick as you can afford) as there is no provision for a mouse. It will autoboot if you have TOS in ROM. Otherwise, you can still run it from the GEM desktop. The joystick plugs into port 1 so you don't have to plug the mouse in unless you plan to play in the two player mode.

Basically, you (as a karate master sporting a pair of nicely tailored blue pajamas) must tangle with any number of tastefully dressed, red-clad foes. These fellows, of course, prove to be increasingly skillful as time wears on.

In between hand to hand bouts of fist and footisticuffs, you are also required to pulverize counters groups of mystically bouncing clay pots. This happens every four rounds and is supposed to be some sort of "purity" testing. In most cases it simply proves to be a hassle. You have a time limit in which to break all the pots or lose one of your men. I much prefer a bonus situation where you don't make it, but you don't lose any men either. There's nothing more frustrating than fighting your way up to a very high level and then losing your belt to a gang of

hyperactive clay containers.

After defeating your first four single opponents and smashing your first set of porcelain adversaries, you must challenge two opponents at a time throughout the rest of the game. Needless to say, batting two scarlet-clad assailants is roughly twice as hard as battling just one. Periodically, however, one of those wildly bouncing clay pots will show up on the screen. If you can nail it before it disappears, you gain some of your health back. If you're mixing it up with two of these guys, those pots become very important.

The main thing to keep in mind when fighting two at a time is not to get stuck in between them, as they will bring you to your knees very quickly that way. You have to learn to do somersaults (one of the more difficult moves to master) to keep flipping bodily out of harm's way. Besides that, periodically, little twirling shuriken spin across the bottom of the screen and you must leap over them or lose even more health. A flying jump kick will also get you over this dilemma and possibly rob your opponent of a couple of his front teeth if you time it just right. If, rather than a clay pot, you espy happily vacillating oriental mask, smashing this will earn you not only regained physical health, but indeed another life!

ST Karate also includes a two player game mode. This is often the most challenging way to play the game because you confront your partner one-on-one as opposed to handing the joystick back and forth. It's three fails, no time limit, and that's it. There isn't even any score. It's simply up to you to outjump, outpunch and outkick your opponent. You still have the opportunity to glean extra health points and lives along the way...if you move fast enough.

The other great thing about the two player mode is that if you play it by

MAC INKER REVIEW
by Marcy Milsop

I don't know about you, but I use my printer ALOT! My husband and I run our own business, I am involved in Cub Scouts and of course our computer club. I also do work for a company that does property reports which require printouts of about 500 pages per month. As you can see my printers get alot of use. This also means that I go through quite a few printer ribbons, accumulating quite an expense

In going through the Computer Shopper a couple of months ago I came across an ad for a printer ribbon reinking machine. I had heard that these machines were quite expensive but this one listed for $68.50 and claimed to reink any type of cartridge ribbon. It was called the Universal Mac Inker produced by Computer Friends, Inc. 14250 N.W. Science Park Drive, Portland, Oregon, 97229, 1-800-547-3303.

Knowing quite a few people with printers I figured I could probably reink their ribbons for a dollar or two each and help pay for the machine.

After calling Computer Friends, I was a little disappointed. Although the Mac Inker will work with any ribbon cartridge, you must purchase a driver for each cartridge type you use. The cost of the drivers were $8.00 each, and at that price, I discovered I could not purchase drivers for all printer types. After doing some calculations I discovered I could recover the cost by only reinking my own ribbons in about 9 months. I decided to go ahead and order it.

I placed the order on December the 19th and after several additional phone calls I received a package on January 14th. I suppose Christmas had something to do with the delay so I wasn't too upset.

After unpacking the box I found myself a little confused about putting the machine together. The picture in the instructions didn't quite match what I had in my hands. I am fairly competent with computers but when it comes to mechanical things I am a complete idiot. I waited until my husband, David, got in and after he did some further reading he discovered that the design had been updated and a additional instruction sheet described the changes. Finally the Mac Inker was together, plugged in and ready to try out.

Turning the switch on I discovered how SLOWLY the motor turned. The ribbon cartridge is placed on the driver and wrapped around an ink reservoir, as the motor turns it pulls the ribbon across the reservoir and spreads a thin line of ink across the ribbon. I had thought it would take a couple of minutes to reink a ribbon. WRONG! One pass of the ribbon took about 12 minutes. The literature accompanying the Mac Inker said that one pass was sufficient and that most problems were caused by over inking. WRONG AGAIN! It took about 3 passes around, approximately 30 minutes, to sufficiently reink a ribbon.

Clean up is a little messy also. The ink reservoir must be emptied, and the guide rings removed and cleaned with WD 40 or LECTRA CLEAN. A rubber cover is provided so the reservoir does not have to be spotless, but as you can see this is a messy task.

Computer Friends Inc., claimed in there advertising the Mac Inker would 'Dramatically improve print-out quality' and this it certainly did. The printout was much darker and clearer than even NEW ribbons. The ribbons even last longer, and by ordering additional ink reservoirs, you can reink color ribbons. ($2.00 each)

The Mac Inker would be an excellent buy for a company or someone who uses many ribbons a month, but I really can't recommend it for the average home user. I think I will get my money's worth, however, I wish the process was a little faster and not quite so messy.

A Null Solution

by Tony Lee

After buying my new 1040ST system, I started wondering what the heck was I going to do with my old 8-bit system? Another question popped up and that was 'How was I going to transfer those important letters and documents over.' Well, this leads us into this month's subject and that is file transfers between the 8-bit and the 16-bit Ataris.

The solution to this problem is quite simple and that is to construct a null modem. A null modem is basically a cable that hooks up the data output of one computer to the input of another. The use of the null modem requires that you have the following system configurations:

8-bit:
A 400/800, XL/XE
Disk drive
850 interface or other serial port interface
A terminal program like AMODEM 7.1

16-bit:
ST computer
Terminal program

The construction of the null modem is straightforward and only requires some simple common parts and soldering skills. Now let's begin the construction of the cable. First go out and purchase the necessary parts:

1 Female DB-25 connector (Amphenol is recommended)

1 Male DB-9 connector (Amphenol is recommended)

6' of 3 conductor shielded cable

The reason why I recommend that you purchase the more expensive Amphenol connectors is because the Amphenol allows you to remove the pin contacts. This feature is especially useful since we would not like to make a new cable just to do file transfers with the Mac for the Magic Sac Cartridge. By changing the pin housing, the same cable can also be used with computers that don't have DB-9 connectors. The second reason that the Amphenol connectors are preferred is that it is easier to make the solder connection with the pins removed.

Now that we have our parts, we can now construct the cable between the computers. The following pins should be connected together to complete the cable:

DB-25 (ST)		DB-9 (8-bit)	
2	Transmit Data	4	Receive Data
3	Receive Data	3	Transmit Data
4	Request to Send	8	Clear to Send
7	Ground	5	Ground
8	Carrier Detect	7	Request to Send

Now that the cable has been constructed, we can proceed to do some data transfers. Within our club's PD libraries we have the transfer programs that Analog used with their null modem. I don't like to use them mainly because they do not provide for adequate checking for errors, and using this method we can transfer files from the ST to the 8-bit Atari. By using some of the many available terminal programs, this is no longer a problem.

For the ST I chose to use ST Talk by QMI and for the 8-bit I used AMODEM 7.1. With the cable now in place, the transfer was quick and easy. First, both of the terminal programs must be set to the same baud rate (the higher it is, the quicker the transfer will be). Second, ready the ST to receive the file. Third, have the 8-bit send the file.

The null modem is not limited to this particular hookup, but instead you can use this method to transfer ASCII data from other machines.

And Software Toolworks Too

The gang from Software Toolworks will be at our general meeting to show off Chessmaster 2000 for both the ST and XL line of machines.
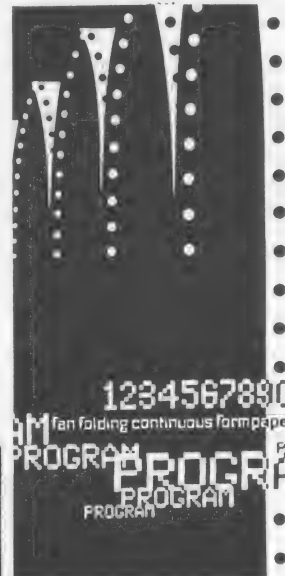
THE OLD MAN AND HIS COMPUTER
Live and Learn

Current note 2-87

by Robert W. Ford

I was fifty-five years old when I bought my first computer. That was in December 1985. Not having used one before, it was a most difficult decision for me. This article is about my experiences of the past year, some of my mistakes and other thoughts. Maybe it can help as you also live and learn with a new computer.

One of my greatest joys has been teaching my six year old granddaughter to use the "com-poot-er." I have DEGAS, a graphic arts program. One time I asked her if she would like to draw on the computer. "Yes Grandpaw," she said, "let me go get my crayolas."

For a couple of years I was under pressure from my boss to learn to use the computer at the office. My view was, "If you do it yourself, you can't supervise it." I did not think that I needed a personal computer at home just to balance my checkbook.

A cousin got me interested in doing family genealogical research. I began to accumulate a lot of information and to write a lot of letters. I purchased an electronic memory typewriter - my first mistake. The data base became to large for me to manage by hand. I needed help, and the time had come for me to get a computer.

So, being true to my managerial training, I began to look at some computers, read material, compared prices, studies specifications, and talked to others with PC's. I knew that I did not want an IBM or IBM compatible because I had no intentions of doing office work at home. Why ask for trouble when I had a boss who was a computer (nut) enthusiast? I was not going to make that mistake!

After considering all the facts, my choice came down to either the Atari 130XE or the 520ST with color monitor. I decided on the ST because of the capacity for the price and the new 16 bit technology.

I never seem to do anything for myself without going whole-hog! I got a Panasonic 1091 printer, Volksmodem, computer table, cables, interfaces and a surge protector. The software included VIP Professional (a Lotus 1-2-3 look-a-like), FinalWord (which I do not use) H&DBase, two spelling checkers, games and more, all of which I purchased. I now have an interface for the typewriter and a lot of supplies with over 150 disks. The wife says all this was the real mistake.

My next mistake was buying another single sided disk drive because I got so tired of swapping disks when making a backup copy. I now wish I had purchased a

double sided drive with 1 MB. I did get a TOS ROM chip installed.

I began to teach myself the computer and to use the software. The manuals assume a level of knowledge that as a novice I did not have. Learning was slow and difficult and took a lot of time and effort. I learned Lotus 1-2-3. My mistake there was letting the boss know. Now sometimes I do office work at home, but mostly I do personal work.

I have written more personal letters this past year then all the rest of my life. I find that enjoy writing now. I have also learned that an unhappy consumer with a computer can become a nuisance to a business slow to resolve a consumer problem.

In my zeal to learn as much as possible, I subscribed to four publications. They were Current Notes, Personal Computing, Compute and Analog. As soon as Current Notes and Analog arrive, I read them from cover to cover. The other two magazines are not much help. The ST SIG of Novatari is most helpful even if I am not a programmer.

I enjoy reviewing bulletin board files and information and have now learned to download what is of interest. I have not learned to upload, but that does not matter. As a novice, I really have nothing to contribute - as yet.

It took me several months to learn to manage my files. It seems I forgot the first rule of management - that is, planning. Without some forethought about record keeping, you waste a lot of time. I have now arranged my disks by both subject matter and software used. Each program now has a boot disk set up for its purpose and my particular needs for that program. Many of my files have dates of November 1985 because I would forget to set the date and time from the control panel. Now all boot disks have Biclock and Mousetrap in the Auto folder. I have to set the date before I can open a root-directory. I have purged all working disks of any unnecessary files to save space.

The 512K is to small for my genealogy data base. What mistake will I make next? Should I get the 1 Meg upgrade? Will I keep buying software? Get and internal clock? You live and learn. You should profit from your mistakes or some dealer will.

SERVICING THE XL/XE
POWER SUPPLY

MACE
2-89

by: D.F.Neff

This article is a step-by-step trouble-shooting guide and parts list to permit you to repair your own computer power supply. It is written for the Atari enthusiast who is not well-versed in the electronics field, but has some interest in learning more. It assumes you are familiar with soldering techniques and the use of a VOM. Those of you who are hesitant to repair anything yourself should remember two things:

1.) It's already broken, you can't make it worse.
2.) Nothing in the power supply costs over $10.00 to replace. In fact, the parts most likely to fail cost less than $1.00!

Now let's get started. First, we'll consider the obvious questions. Is the power supply plugged in? Is the wall outlet working? Plug a table lamp into the wall outlet to test it. If you are using an extension cord between the wall outlet and the power supply, plug the light into the extension cord to test it too. If the outlet and the extension cord pass that test, we can turn our attention to the power supply.

Plug the power supply back into the wall outlet and remove the power supply plug from the keyboard console. This plug will contain seven pins as shown in Figure 1. The pins are connected in two groups of three with no connection to the middle pin. Using your voltmeter, check for the presence of 5 to 7 volts d.c. between the three pins on the left and the three on the right.
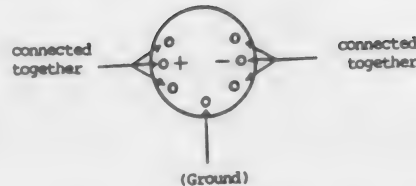
Figure 1

If this plug has the proper voltage and polarity then your power supply is working and your problem is located inside the keyboard console. If, however, you do not have the proper readings at this plug then you must do further testing inside the power supply.

Turn the power supply upside down, gently pull out the four rubber feet and unscrew the Phillips-head screw under each foot. Turn the power supply back upright and remove the top half of the case to expose the circuit board and components.

Remove the fuse and test it with your ohmmeter. If the fuse is O.K. reinstall it and proceed with the test program. If the fuse is blown and the replacement also blows, go to the test description for CR101-64.

13

The cord to the keyboard is attached to the end of the circuit board at points labeled GND and +5V. With the power supply plugged into the wall outlet, use your voltmeter to test for 5.0 to 7.0 volts d.c. across these two points. If the proper voltage and polarity are present between these points then your console cord and plug are defective and must be replaced. When you have replaced the plug and cord, use your voltmeter to check for proper polarity as shown in Figure 1, before hooking up the keyboard console again.

If no voltage was present at those two points we must go to the opposite end of the circuit board and test the transformer. The two secondary leads from the transformer attach to the circuit board near the fuse we tested earlier. Set your voltmeter for a.c. and check for 10 to 15 volts a.c. between these two leads. If no voltage is found here we must check the primary side of the transformer.

Notice: This next step requires working with dangerous voltages! If you are not experienced in handling 110 v.a.c., stop here and take your power supply to a repair shop. If you wish to proceed with this test, unscrew the two wire nuts attaching the power cord to the primary side of the transformer. Use your voltmeter to test for 110 v.a.c. across the two bare connections. If there is no 110 v.a.c. at this point, the power cord and plug are defective and must be replaced. If you find 110 v.a.c. at this point, then the transformer is defective and must be replaced. The transformer listed in Table 1 is is not a direct replacement and may present a slight fitting problem when you reassemble the power supply case.

If, when you checked the secondary wires near the fuse, you found the 10-15 v.a.c. to be present you should skip the 110 v.a.c. test. Instead locate [JC] U102 which is attached to the black finned heat sink. It will look like Figure 2.

14
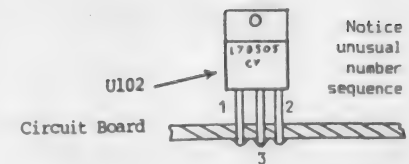
Figure 2

Remember when we first opened the case and checked the attachment points for the keyboard console cord and plug? Those two points are labeled GND and +5V and are located on the circuit board where the two wires enter the case. Attach your voltmeter's black or negative wire to the point labeled GND and leave it there while we test U102. Set your meter for d.c. again and touch the red or positive lead of your meter to each of the three legs of U102 one at a time. Be careful not to short two legs together while doing this test.

Pin 3, the center pin, should have about 1/4 volt on it. If it is lower than 1/4 volt it's o.k., but if it is higher than 0.5 volt, R102 may be defective. R102 is a 2.9 ohm resistor but can be replaced with three parallel 10 ohm resistors.

Pin 1 should have 11 to 16 volts on it, and pin 2 should have 5.5 to 7 volts on it. If pin 1 and 3 voltages are normal, but pin 2 voltage reads wrong, then U102 is defective and must be replaced.
If the voltage at pin 2 is too low, the problem is in the current rectifier diodes CR101-4.

Unplug the power cord from the 110 v.a.c. wall socket and allow the power supply to sit idle for at least two

minutes to allow C104 to discharge before proceeding. Set your meter to read resistance.

Check the resistance of diode CR101 and write it down. Reverse the meter leads (use the red where you just used the black and vice versa) and write down the new resistance reading. The value of these readings is not critical, but one must be at least twice as large as the other.

Repeat the same procedure for diodes CR102, CR103, and CR104. If any one of them fails this test, replace all four of them. The failure of one diode places excessive loads on the others and may lead to their failure later. Before removing a diode for replacement note the orientation of the silver paint band on one end of the body. The replacement diode must be installed the same way. If you get mixed up, refer to the diagram on the circuit board and refer to Figure 3.

Figure 3

Now look at the other components on the board. If any of them are damaged or appear burned, replace them. At this point your power supply should be working as good as new. Use a small brush to remove any dust in the case, and reinstall the cover.

15

PART DESCRIPTION		RADIO SHACK STOCK#
Voltage Regulator	U102	276-1770
2.9 ohm Resistor	R102	271-1301 #
Power Transformer		273-1511
Console Power plug		274-003 ##
Rectifier Diodes CR101-4		276-1143
Fuse		270-1246
Power cords		278-1255

#(use 3 of these connected in parallel)
##(not a 7-pin plug but it will work well)

Table 1

MACE ONE LINER BY NEFF
10 POKE 755,4:FOR T=1 TO 600:NEXT T:POKE 755,0:FOR T=1 TO 600:NEXT T:GOTO 10

AQUARIUS

THE WATER

BEARER

PROGRAMMING LANGUAGES FOR THE ATARI ST
by Steve Marshall

Part 4: "C"

In the good old days of the 8-bit computers programming languages came in two flavors - Basic and Assembly. Oh sure, you could do some things with Logo, Forth and some of the more esoteric languages of the day, but when you picked up any of the computer magazines what you saw were listings in Basic or columns of cryptic numbers and letters that when typed in yielded a "machine-language" game. And since few of us were willing to tackle assembly, Basic was the language we all cut our teeth on. But now you have an ST, one of the new breed of powerful computers, and suddenly all you hear about is this mysterious language called "C".

C was invented in the early 70's by Dennis Ritchie at Bell Laboratories (AT&T) while in the process of developing a new operating system called UNIX. C was used to write the C compiler, the UNIX system itself, and a wide variety of programming tools. Since that time, the UNIX operating system has gained widespread popularity. And as the use of UNIX spread, so did the use of C. Microsoft picked up on C as the primary development language for all of its software including the MS-DOS operating sy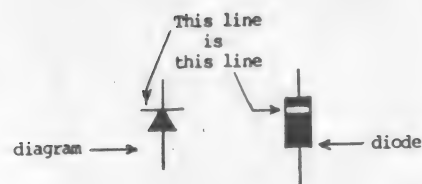stem. And, of course, GEM on the ST is written almost entirely in C. Why the popularity of C? There are several reasons. C is a high-level language but, unlike most others, C is relatively close to the machine language level in how it works with numbers, memory addresses and programming structures. For example C, unlike or other high-level languages, allows bit level manipulation using left and right shifts, bitwise AND, OR and EXCLUSIVE OR. C is also considered an "easy assembler" because compiling a C program generates pure machine code and enables programs to be written up to ten times faster than programming in assembly language. It is a very compact language with about the same number of language elements as Basic (around 30). Besides its speed, compact syntax and ease of use, the portability of C is of primary interest to programmers. Portability simply means that programs written in C on one computer can be quickly and easily ported to other computers with a minimum of effort. Many professional software developers consider C to be the programming language of the future.

C is similar to Pascal in many respects. It too is a structured language based on the use of small sub-programs called "functions". Each function is a self-contained program that performs a particular task. A sample C program always contains a function called main() that calls other functions in sequence. By breaking down a complex task into small, manageable tasks, programming effort is simplified. Most C compilers also contain libraries of functions that can be used by the programmer for simple tasks like printing a character to the screen or disk I/O. These libraries enable the C programmer to avoid "re-inventing the wheel" each time a program is written by allowing the program to "include" the appropriate library functions during the compile and linking process. This type of modularity makes it easier to write

and maintain complex programs.

A C program is written using a common text editor just as any text document would be prepared. After the program has been written, it is compiled using a C compiler. This compiling translates each function into the machine code understood by the computer. Errors in syntax are usually caught at this stage. When the program compiles successfully, it is then linked with the various libraries it references resulting in a finished, executable program. One of the advantages of C is that it produces "stand-alone" programs that do not require any support programs or runtime packages. On the ST, its just a matter of "double-clicking" on the icon to run a C program, whereas Basic programs require loading the language from disk before running the program. There are several good C compilers available for the ST at this time. The original C compiler, Alcyon C, was included with the developer's kit and by virtue of its being the first is still considered the standard. Unfortunately, it is no longer the best, having been superseded by the excellent Megamax C and the newly released Mark Williams C. For those interested in learning C I recommend that you read the numerous reviews of the available C compilers. My observations seem to indicate that Megamax C has gained the upper hand among programmers with Mark Williams gaining support particularly among the more experienced.

For those of you interested in programming in C, keep in mind that, though C has a more cryptic style and syntax than Basic, it still incorporates the basic concepts of conditional branching (IF-THEN-ELSE), loops (FOR-NEXT, DO-WHILE), and general programming conventions. If you've learned to program in Basic, either on the ST or on an 8-bit machine, then the switch to C should not be too painful. There are a number of books on the market to help you including the "bible" of C programmers, "The C Programming Language" by Kernighan and Ritchie and a book I found useful in making the transition, "ST Basic to C" from Abacus Books. Another help is to study the various articles on C programming that appear in virtually every ST magazine. In particular, Clayton Walnum's series, "C-Manship", appearing monthly in Analog magazine is geared to the beginner.

Learning to program in C, and especially learning the ins and outs of programming GEM on the ST, will be challenging and sometimes frustrating but the rewards will be great if you persevere. Don't be afraid to seek help through your Atari user group. An ST programmer's SIG meets monthly and would welcome your input and support. Next month, we'll take a look at two of the oldest computer languages, Fortran and Cobol.

ATARI SCUTTLEBITS
Hacking the ST Mac

by Bob Kelly

I intend to start off the New Year by discussing Data Pacific's Macintosh Cartridge (*Magic Sac*). It is NOT my intention to provide a detailed technical review of the product but to provide insights on "true" hardware set-up cost, evaluate whether the *Magic Sac* can be considered a full Mac clone, supply a list of software considered essential, and to provide other recommendations/hints so others may avoid my learning curve frustrations.

COST OF SYSTEM SET-UP

The Mac cartridge (hereafter called the *Magic Sac*) fits into the cartridge slot of the ST. The discounted price for the *Magic Sac* I is about $90 while the *Magic Sac Plus* runs about $120, the difference being inclusion of a clock module on the *Plus*. In addition to the cost of the cartridge, two Apple Mac ROMs must be purchased and inserted into the cartridge. The cheapest price I have seen for the two ROMs is $30 from B & C Computers in Santa Clara, California. (More often than not the ROMs are advertised at $50 per pair.)

In addition to the DIRECT cost of the Magic/Mac cartridge, the INDIRECT cost associated with modifying your ST's hardware configuration must be considered.

First, for best viewing, a monochrome monitor is required (remember, the Mac has no color). Data Pacific, using clever programming techniques, now allows a color monitor to be used (beta test version on *Magic Sac* boot disk 3.5). However, the picture quality, in diplomatic terms, is very unpleasant for those of us with normal eyesight.

Next, your Atari ST should be equipped with 1 megabyte of RAM for best operation. If you have standard 10S in ROM on a 520ST (512K), the largest Mac emulation possible is 256K. This memory size restricts the flow of operations by causing more disk swapping and some unexplained happenings (e.g. system crashes). In addition, the COLOR MONITOR ONLY OPERATES with a 1-Meg machine at present.

If you do not want to do a considerable amount of disk swapping, it is best to have a two drive system. A one drive system promotes tennis elbow, similar to CP/M 2.2 operations.

Since I own a 1040ST with two drives, I did NOT have to purchase indirect items other than a monochrome monitor ($125). The total hardware cost of my set-up was approximately $250 ($125 + $90 + $30 + shipping)

A FULL MAC CLONE — NOW OR EVER?

What did I get for this $250? First and foremost, I learned how a Macintosh operates. I would buy the *Magic Sac* again for this alone. However, for those who have a Mac at work, the *Magic Sac* is NOT NOW a usable clone. It is a system best suited to users who can be classified as quasi-hackers or those who want to learn in depth about the Mac. On what do I base this opinion? The *Magic Sac*:

- Does not support the Epson/Star Micronics/ Panasonic array of dot matrix printers (this deficiency is a result of Apple's corporate policy, but it is now a technical problem for the *Magic Sac*). Almost everything in Mac public domain looks for the Apple Imagewriter. Data Pacific (DP) plans to have an Epson driver in version 4.0. When will 4.0 be available - my guess, March, 1987.

  ASCII files can be printed with the aid of some Mac public domain printer programs -- more on this later. Let me hasten to add, very few files found on Mac BBSs are in ASCII format -- MacWrite 4.5 files in binary form are the rule.

- Does not support *MacWrite 4.5*. It will work with *MacWrite 2.2* which saves files in ASCII form. Data Pacific is working on this shortcoming and intends for the *Magic Sac* to be compatible with *MacWrite 4.5*.

- Does not run copy-protected software. As such, *Microsoft Word*, *Pagemaker*, *Excel*, and *Jazz* cannot be run. If you don't know why these programs have been essential to the success of the Mac -- stop reading here. Data Pacific plans to produce a drive capable of running copy-protected programs which may be available in the May/June period.

- Does not support sound. In my opinion, this is not a major liability for the foreseeable future.

SOFTWARE TO GET STARTED

Without doubt this area was and remains the cause of more headaches than any other. A substantial amount of the problems were the result of having to use Magic boot disks 3.0 or 2.0. Thank ????? that is over!

With the introduction of version 3.5 of the Magic boot disk, several of the more persistent problems have been solved -- such as frequent "crashes" when changing programs and bugs in the way *Switcher* runs. Also, *Notepad* no longer sends the system into orbit.

Most importantly, version 3.5 permits Macintosh desk accessories to be installed and removed (unlike Atari, it is not as simple as changing the file extender to .ACC).

Operating with anything less than version 3.5 is just plain stupid. For those who still have versions 2.0 or 3.0, you can download 3.5 from CompuServe. Proceed to the Atari Developers Forum (type: "Go ATARIDEV"). It is an ARC file and 39K in length -- remember the *Magic Sac* boot disk is a normal Atari formatted disk.

Along with 3.5, you should have the following public domain/shareware disks:

A. Finder: ver., 4.1 (system disk)
B. Font/DA Mover: ver., 3.2 (changes fonts and desk accessories)
C. Freeterm: ver., 1.8 (telecommunications)
D. PackIt III: ver., 1.2 (compresses/decompresses files)
E. Ink: ver., 2.0 (prints ASCII files)
F. Chooser: ver., 2.0 (selects print driver)
G. MacWrite: ver., 2.2 (Commercial Word Processor.)

The Finder disk is the start-up or boot disk for the Mac system. On the disk is a system folder containing system files and *Finder 4.1*. *Finder* performs such tasks as designating which disk is in control, provides the entrance and exit to programs (i.e., finding files), and performs general housekeeping and control procedures (copying files).

Font/DA Mover 3.2 permits the installation/removal of both different fonts and desk accessories. *Font/DA Mover* version 3.2 is far and away the best version with the *Magic Sac*. Earlier versions 2.5 and 3.0A.1 do install and remove desk accessories but at a much slower rate. Further, versions 2.5 and 3.0A.1 do not work when installing fonts. (Remember, *Font/DA Mover* works only with Magic boot disk 3.5 or better).

Freeterm 1.8 is more than adequate for most telecommunication needs.

PackIt III is desirable since some of the programs on the Mac BBSs are compressed. *PackIt III* will decompress these files.

Now comes the question of printing. Again, Magic boot disk 4.0 is soon to be released and designed to solve many of these print driver problems. If you want to print out a document now, some precise procedures have to be followed. First, click on PARALLEL PORT when booting. Next, for binary files compatible with *MacWrite 2.2*, select and load the document you want into *MacWrite* and save it as an ASCII file (use: Save as ..... text only). Quit *MacWrite* and boot up *Ink*. Pull down the printer menu and click on Printer Port and Okidata 92 (other printers can be configured -- read help files). Next, go to File Menu and click on Print

Selected Files, now:

-Click on OK box (don't enter anything)
-Click on File to Print
-Click on Select
-Click on Done
 New box appears asking if you are "Ready to start printing"
-Click on OK.
 Your file should now be printed.

Another alternative print method requires a commercial program called *Epstart* ($50 retail). This program supplies several different Epson printer drivers which are placed in the system folder. With the aid of a public domain desk accessory called "Chooser or Choose Printer", the Epson driver is installed as the default driver. Repeated attempts have failed to get it to work. Again, it appears not to be solely a fault of the *Magic Sac* system. A local Apple dealer has stated that it works so infrequently with the Apple Macintosh they no longer carry the product.

RECOMMENDATIONS/OPERATIONAL HINTS

When using the *Magic Sac*, I suggest:

1. Never use the Motivator (RAMDISK) unless you have become thoroughly familiar with its operation. Simply put, it keeps dumping the same contents of RAM to subsequent disks. Suppose you decide to eject the initial disk and the need arises to eject the second disk, it will write back the contents of the first disk when ejecting so ..... 'bye-'bye second disk.

2. Now that you will never use Motivator, do not write-protect the disks by closing the notch on the disks. By physically write-protecting, the *Magic Sac* automatically locks all programs. They cannot be unlocked without stopping work, ejecting the disk and opening the write-protect tab.

3. Rumor has it that the old ProWriter is identical to the ImageWriter I. Thus, according to hearsay, the printer driver problem would go away if you own one.

Recommendations for David Small/Data Pacific:

1. Please, in preparing printer drivers for the *Magic Sac*, make it possible to meet the requirements of the Star Micronics and Panasonic printers. Why do I say this? An extensive survey involving hundreds of Current Notes readers indicated that about 45% of all Atari owners have Star or Panasonic printers.

2. In booting the *Magic Sac* system, make the default printer port parallel, not serial.

3. Unless the color monitor system can be substantially improved, it will only dissuade serious applications.

③③

4. A very few programs do not work with Magic 3.5 that worked with 3.0, notably Desk Acc. Mover, ver., 1.3/4

5. In the design of the forthcoming Gem based formatter/copier, the most needed addition is the ability to repeatedly format or copy disks without having to go through all the prompts or rebooting the program.

6. It would be nice if some of the more feature-laden telecommunications programs could be used with the Magic Sac, in particular, I refer to auto dialing capabilities, kermit protocol, etc.

7. In my opinion, the ultimate success or failure of this system will depend on the introduction of a low cost Macintosh disk drive capable of running copy-protected software (this assumes the printer driver issue is satisfactorily resolved and soon).

Given the positive response to user input to-date and timely product updates, the Magic Sac will probably approach full clone status by summer. Transporting work between home and office should then be a reality. It is not too early to start learning. In that vein, the best book I have used regarding Mac operations/software is The Macintosh Advisor by Harriman and Calica, (Hayden Co., 1986).

I would like to express my appreciation to Mr. Jeff Greenblatt for his expertise. All the public domain/shareware that he and I gathered for this article has been given to the Current Notes Mac library.

CLASSIFIED ADS

---

③④

Comparing ST Word Processors

Largely a Matter of Personal Choice

by Milt Creighton

THIS MONTH: ST Writer, 1ST Word, Boffin, Text Pro, HippoWord, Word Writer ST, and Regent Word II.
NEXT MONTH: HabbaWriter II, Boffin (updated), and (possibly) MicroSoft WORD.

Like underwear and automobiles, word processors are largely a matter of personal choice. We buy word processors for all sorts of reasons, but the program we will use most often is the one we are most comfortable with. For most of us, that means the word processor we know best (possibly the first one we learned and the only one we know), the one we hope will cause us the least amount of trouble. Word processing is supposed to be a timesaver, after all. If we find ourselves constantly forced to pour over manuals while we search for obscure commands with which to perform simple but necessary functions, it just isn't worth it. Better to get out pencil and paper and go back to the dark ages.

Early word processors for the ST tended to be fairly basic in their design if they utilized GEM (like 1st Word). Others were ported over from other machines (such as 8-bit word processors like ST Writer or 16-bit translations like Final Word). On the one hand, the early GEM-based programs had a certain elegance and grace but little sophistication while, on the other hand, the translations were often powerful but clumsy by comparison. Since both 1st Word and ST Writer were either bundled with ST computers or could be had for the asking, there seemed little reason for most of us to buy the early commercially offered programs. A waste of money, we thought, unless a particular favorite had been ported over to the ST.

THE BIG CHOICE

Now there are more than a dozen word processors available for the ST line and more on the way. The newer ones are both GEM-based and have a good deal of power to offer as well. On-screen formatting, display of multiple typestyles, and a complete array of block operations between multiple files are available in many. A few still use a preview screen for viewing the final version which permits the user to see the formatting codes on the work screen, but this approach doesn't appear to be as popular in the newer products. Some of the new offerings also permit the integration of graphics and text within a single file -- certainly one distinction of second generation ST word processors.

One of the disadvantages of having so many word processing programs on the market is the difficulty of making a choice between them, especially if there isn't

time or opportunity for research. That's where this article can be useful. It compares five of the newest ST word processors to more than 80 criteria and presents the results in tabular form. The first two entries in the table are ST Writer (version 1.50) and 1st Word (version 1.06). These latest versions of "freeware" should be viewed against the commercially available word processors, especially in the areas which are most important to you, before you decide to go out and spend your money.

The criteria used in the table include some that Ian Chadwick used in his article in the second issue of START magazine but I have eliminated others and added some of my own. In addition, with a few important exceptions, whenever all of the word processors met or failed to meet a certain criterion, that criterion was eliminated from the table. The purpose of this article is to point out the differences between the compared programs, after all, not the things they all had in common. I deliberately have not made any attempt to rank-order the entries in any fashion. To do so would be to insert my own prejudices into what is essentially a personal choice; the perfect word processor for me might be ill-suited to meet your own needs.

The commercially offered word processors compared in this article include Text Pro (see full review in this issue) by Abacus Software, Boffin from Software Punch in the UK (not yet available in this country although an updated version will reportedly be offered for sale here), HippoWord from Hippopotamus Software, Inc., Word Writer ST by Timeworks, and Regent Word II from Regent Software.

All of the newer word processing programs emphasize certain areas which the author(s) believed important. Consequently, each will have relative strengths and weaknesses vis a vis the others. Boffin, for example, heavily emphasizes its graphics capability, but less attention is paid to block operations or certain editing functions. HippoWord has very powerful "search and replace" operations, in fact, they are more extensive than I've ever encountered before. In addition, it has the ability to integrate graphics and text -- but you had better make sure your printer is supported or you will find yourself shelling out another $49.95 to create your own printer driver (assuming you know how to alter C code). As other examples, Regent Word II is the only one to offer microdot justification and Text Pro should be a favorite of C programmers because of its automatic formatting capabilities. Finally, Word Writer ST is an excellent all-around first generation ST word processor because of its power and ease of use and it has the best written manual of the lot.

WORD PROCESSOR COMPARISON (35)

Feature	ST Writer	1ST Word	Boffin	Text Pro	Hippo Word	Word Writer	Regent Word2
Cost	Free	Free	?	$49.95	$89.95	$79.95	$99.95[1]
Copy Protected	No	No	Yes	Yes[2]	No	No	Yes
SCREEN DISPLAY:							
GEM-based	No	Yes	Yes	Yes	Yes	Yes	Yes
Number of Files in RAM at one time	1	4	1	1	1	4	1
HELP screens available	No	No	Yes	No	Yes	Yes	Yes
What-you-see-is-what-you-get	No	Yes	Yes	No	No	Yes	Yes
Integrated text and graphics	No	No	Yes	No	Yes	No	No
Display multiple typestyles on-screen	No	Yes	Yes	Yes	No[3]	Yes	Yes
Display multiple fonts on-screen	No	Yes	Yes	Yes	Yes	No	No
User-definable fonts	No	No	Yes	Yes	Yes	No	No
On-screen justification	No[5]	Yes	Yes	No	Yes	Yes	Yes
Shows page breaks on-screen	No[5]	Yes	Yes	Yes	Yes	Yes	Yes
Displays current page number	No	Yes	Yes	Yes	No	Yes	Yes
Capslock on/off indicator	Yes	No	Yes	No	Yes	Yes	No
COUNTERS							
Column counter	Yes	No	Yes	Yes	No	No	No
Word counter	No	No	No	No	Yes	Yes	Yes
Line counter	No	No	Yes	Yes	Yes	Yes	Yes
Page counter	No	No	Yes	No	Yes	Yes	No
Remaining RAM counter	Yes	No	No	No	Yes	Yes	Yes
CURSOR MOVEMENT							
Move cursor to line/page number	No	No	Yes	Yes	No	No	Yes
Page screen up/down	Yes	No	Yes	Yes	No	Yes	Yes
Move cursor word by word	Yes	No	Yes	Yes	Yes	Yes	Yes
BLOCK OPERATIONS							
Move block/copy block	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Cut & paste block	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Cut and paste between screens/documents	No	Yes	No	No	No	Yes	No
DELETE OPERATIONS							
Delete line	Yes	No	No	No	Yes	Yes	Yes
Delete word	No	No	No	No	No	Yes	No
Delete block	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Delete to end	Yes	No	No	Yes	No	Yes	No
Delete file from disk	Yes	Yes	No	Yes	No	Yes	No
PRINT FUNCTIONS							
Permanent printer driver selector	Yes	Yes	Yes	Yes	Yes	Yes	No
Can send special printer codes	Yes	No	No	Yes	No	Yes	Yes
Print only selected pages	Yes	Yes	Yes	Yes	No	Yes	Yes
User definable printer drivers provided	Yes	Yes	Yes	Yes	No[8]	Yes	Yes[9]
Page wait command	Yes	Yes	No	Yes	No	Yes	Yes
Print document from edit	No	No	Yes	Yes	Yes	Yes	Yes
FORMATTING							
Alternate headers and footers left & right	Yes	Yes	No	Yes	Yes	Yes	No
Multiline headers and footers	Yes	No	Yes	Yes	Yes	No	Yes
Multiple column capability	Yes	No	No	Yes	Yes	No	No
Variable line spacing	Yes[4]	Yes	Yes	Yes	Yes	Yes	Yes
Forced page break capability	Yes	Yes	No	Yes	Yes	Yes	Yes
Subscript/superscript commands	Yes	Yes	No	Yes	No	Yes	Yes
Indent/outdent feature	Yes	Yes/No	Yes/No	Yes	Yes	Yes	Yes
DISK/FILE OPERATIONS							
Can save an ASCII file	Yes	No	No	Yes	No	Yes	Yes
Accepts other word processor files	Some	Some	Some	Yes	Some	Some	Some
Accepts ASCII files	Yes	No	Yes	Yes	Yes	Yes	Yes
Can format disk from within program	Yes	No	No	No	No	Yes	Yes
Can merge files from disk	Yes	Yes	Yes	Yes	Yes	Yes	No
Can delete files from disk	Yes	Yes	No	Yes	No	Yes	No
Automatic backup of files	No	Yes	No	Yes	No	Yes	No

Feature	ST Writer	1ST Word	Boffin	Text Pro	Hippo Word	Word Writer	Regent Word2
MACROS							
Has macro capability	No	No	No	Yes	Yes	No	No
Uses function keys[6]	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Uses alternate key[6]	No	No	No	No	Yes	Yes	No
User-definable function keys	No	No	No	Yes	No	No	No
SPECIAL FEATURES							
Integrated outline processor	No	No	No	No	No	Yes	No
Integrated spelling checker	No	No	No	No	No	Yes	Yes
Print sideways	No	No	Yes	No	No	No	No
Specify block to be printed on same page	No	No	No	No	No	Yes	No
Integrated label maker	No	No	Yes	No	No	No	No
Integrated drawing program	No	No	Yes	No	No	No	No
Integrated chart maker program	No	No	Yes	No	No	No	No
Change point size of text	No	No	Yes	No	Yes	No	No
Rotate letters on-screen	No	No	Yes	No	No	No	No
Mail merge capability[10]	Yes	No	No	Yes	Yes	No	Yes
Auto hyphenation capability	No	No	No	Yes	No	No	No
Index generator	No	No	No	Yes	No	No	No
Table of contents generator	No	No	No	Yes	No	No	No
Auto C-source format	No	No	No	Yes	No	No	No
Integrated Calculator	No	No	No	No	No	No	Yes
Microdot Justification	No	No	No	No	No	No	Yes
MISCELLANEOUS							
Printed manual/Indexed	No/No	No/No	Yes/No	Yes/Yes	Yes/Yes	Yes/Yes	Yes/Yes
Pages in manual	38	42	39	60	45	142	54
Uses all free RAM	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Hard drive compatible	Yes	Yes	No	Yes	Yes	Yes	No
Uses mouse/keyboard commands	kybrd	Both	Both	Both	Both	Both	Both
Word-wrap on/off toggle[7]	No	Yes	No	Yes	No	Yes	Yes
Can edit >80 columns	No	Yes	Yes	Yes	Yes	Yes	Yes
Insert/typeover toggle	Yes	Yes	Yes	Yes	No	Yes	Yes
Work screen/print preview screen	Yes	No	No	Yes	Yes	No	No
Column marker	No	Yes	No	No	No	Yes	No
Reformat paragraph command	N/A	Yes	Yes	No	N/A	Yes	N/A
Reformat entire document command	N/A	No	No	Yes	N/A	Yes	N/A
Global search and replace functions	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Footnotes to Table: N/A = Not Applicable

[1] The price of REGENT WORD II will be reduced to $79.95 on 15 January 1987.

[2] While TEXTPRO is copy protected, it is possible to copy the program to your hard disk and, as long as the original disk is in drive A when the program is booted, it will run without problem.

[3] Although HIPPOWORD cannot display multiple typestyles on its workscreen, it can on its preview screen.

[4] Earlier versions of 1ST word would not double-space but the most current version does have that capability.

[5] ST WRITER displays page breaks on its print preview screen but not on its work screen.

[6] Word processors that do not have macro capability but are GEM-based can often make use of commercially available macro programs such as ALT or ST KEY. However, if the function or <alt> keys are used by the word processing program, they will not be available for use by any other program.

[7] Typing a large table into a document when the word wrap feature is activated can be a frustrating experience. Those word processors which do not have the ability to toggle word wrap on and off usually permit their right margins to be reset to very wide maximum widths, thus accomplishing the same thing for all practical purposes.

[8] For an additional fee Hippopotamus Software provides a printer driver installation program written in C which must be modified for your printer and then compiled. If you aren't a C-programmer, the driver creator won't be of much help to you.

[9] REGENT WORD II provides a printer driver creator in later versions of the program but it is not made clear in the documentation how to use the installation program. You must NOT access the "choose printer" menu at any time if you want the program to use the driver you have created. However, if you choose the "print" option without choosing a printer type, REGENT WORD II will default to your user-defined printer driver.

[10] Most, if not all, of these programs require a compatible data base or mail merge program in order for this feature to work. ST WRITER includes a basic internal mail merge capability as does TEXT PRO. HIPPOWORD may require HIPPOSIMPLE, WORD WRITER ST requires DATAMANAGER ST, and REGENT WORD II requires REGENT BASE or REGENT WORD II MAIL MERGE (sold separately for $24.95)

(37)

To make proper use of the information presented here, first identify your word processing requirements. Do you intend to use your word processor as a text editor in writing programs? Do you plan to use it to write the Great American Novel? Do you want to use it for business correspondence or do you just plan to write an occasional letter to Mom? Once you've identified your own requirements, take a look at the table and see which program comes closest to satisfying your needs. Then bounce the price of that package off what you get for free with ST Writer and 1st Word to decide whether it's worth it.

It might also be worthwhile to point out that you should insure the specific make and model of your printer is supported by the program you buy. This can be critical if the word processor you are buying has the capability to integrate text and graphics. In this case, close isn't nearly good enough. The graphics drivers have to be written very carefully to support each make of printer and I have found "Epson-compatible"

usually doesn't apply in graphics mode — even with Epson printers. So be sure to check it out before you buy.

Perhaps, like me, you won't find one word processor which will solve all your word processing requirements. If so, this table may save you some money by eliminating the more obvious non-candidates. This won't be the last time you see this table, either. As other word processors are released for the ST machines, they will be compared to the same criteria (and perhaps some new ones as well). Next month the second installment will include an updated version of Boffin (probably the one which will be offered for sale in the US), the latest pre-release version of 1st Word Plus, WabaWriter II (reportedly all the rage in Europe), and possibly Microsoft Write (if it's released in time). In the meantime, enjoy looking over the many features of these recent offerings and you'll see that, in word processing at least, the Atari ST machines are second to none.

POLYDISK
RAM Fantasy or Fact

Review by H. B. Monroe

Polydisk is a .5 meg memory up-grade cartridge for the ST. When I first read about Polydisk I wondered who would need or want an external .5 meg of RAM. Would the ST RAM just lie idle while the Polydisk was being used? Is Polydisk the software developer's dream (fast compiling) come true or the poor man's hard disk?

Experience in using the Polydisk has provided affirmative answers to most of my questions and I believe that a lot of people will find Polydisk to be a welcome and useful addition to their ST. The Polydisk cartridge adds 512K bites of memory to the ST through the cartridge port. This means that on the standard 520ST you have the use of a lightning fast RAM disk up to about 1 meg in size. The ST 1040 or an enhanced ST 520 will be able to use up to about 1.5 megs in the RAM disk. A RAM disk larger than 503k, takes the additional RAM from the computer memory. Each RAM disk uses 9k of memory so the RAM disk is always 9k less than the amount of memory used. Therefore the RAM disk is limited in size by the total amount of resident ST Ram plus the Polydisk Ram minus the memory used by TOS and minus the memory used by the RAM disk. After a computer crash press the reset button to fully restore, without loss of contents, a ramdisk setup for 503K or less.

According to Fran Sabolich of the Polyware ST company, the people who are currently useing Polydisk are mostly software developers who need the Polydisk speed, which is superior to a hard disk, and the added RAM. Fran says that handy owners can increase the size of Polydisk to 2 Megs.

Suitable software and instructions come with Polydisk.

I think that many will want the speed of polydisk and will find the price far more acceptable than the price of a hard disk.

The only drawback (a whopper) to Polydisk is that the RAM disk is wiped clean each time the computer is turned off. This means that you must copy to the RAM disk all the programs that you want to use each time you turn on the computer. Of course this can be done automatically with the autoload feature of the ST and may not be too much to pay for the blinding speed and added memory of Polydisk.

The Polyware Company is now developing a battery for the Polydisk which will hold the memory for about five hours after the computer loses power. Sabolich says that the software developers who are currently using the Polydisk are requesting the battery addition to protect the work in progress in case power failure occurs or the computer crashes.

Polydisk Clock and Poly Battery Backup are optional accessorys that can be added to Polydisk.

Polydisk, $199.95. Polyware ST, 5715 Horning Rd., Kent, Ohio 44240. (216)-673-5591 (call after 16:00).

JEWEL OF THE ST
(reprint: Nybbles & Bytes, Jan. 1987)

GEM. The Jewel of the ST. I'm going to expand the scope of this column a little and include some examples in Pascal and C along with Basic. Unfortunately, there are some things Pascal can do Basic won't, and some things C can do Pascal can't.

Here's a short little goody which can be done in all THREE languages: Change the mouse arrow to a pointing hand.

BASIC
```
10 a# = gb
20 gintin = peek(a# + 8)
30 poke gintin,3
40 gemsys(78)
```

C
```
# include = osbind.h =
# include = gemdefs.h =
int dummy;
int work-in[11], work-out[57];
int handle;
int contrl[12], intin[128];
int ptsin[128], intout[128], ptsout[128];
int gr-hwchar, gr-hhchar, gr-hwbox, gr-hhbox
main()
{ appl-init();
open-vwork();
graf-mouse(POINT-HAND, &dummy );
Cconin();
v-clsvwk(handle);
appl-exit();
}open-vwork()
{ int i;
for (i=0; i less than 10; work-in[i + +] = 1);
work-in[10] = 2;
v-opnvwk(work-in, &handle, work-out);
}
```

PASCAL
```
PROGRAM pointing_hand ;
   CONST
   ($I gemconst.pas)
   TYPE
   ($I gemtype.pas)
   VAR
   dummy : integer ;
   ($I gemsubs.pas)
   BEGIN
   IF Init_Gem greater than = 0 THEN
   BEGIN
   Set_Mouse( M_Point_Hand ) .
   Exit_Gem ;
   END ;
   END.
```

ACEED 2-87
— David Lindsley

COMPOSITE ST
Add Composite Video and Audio Jacks to Your 520ST
(Ron Robinson, St. Louis, Missouri - December 1986)

Did you buy a 520ST with a monochrome monitor? Do you wish you could least get a peek at all those neat graphics demos? Do you have a color composite monitor left from your 8 bit days you want to use with your 520ST? Or do you want to connect the sound output of your ST to a good sound system?

The good news is the 520ST includes a large variety video output signals allowing connection to an analog RGB monitor, high resolution monochrome monitor, composite monitor or a TV set. The bad news is Atari decided to bring these signals out to a 13 pin DIN connector which is very difficult to find, and it only allows the connection of one monitor at a time. The good news is it is not too tough to add a couple of common RCA type phono jacks to your machine, simplifying the connection of your 520ST to a composite video monitor or an audio system. An additional idea I will describe is the addition of a switch to allow you to select your composite or monochrome monitor without having to unplug either monitor.

First of all, the warnings. The late model 520STs, which includes 520STs with the built-in RF modulator for connection to a TV set, have a composite video signal brought out to pin 2. If your 520ST does not have the built in RF modulator, you probably do not have composite video available. This modification, while not very difficult, voids the warranty on your machine, and has the ever possible risk of destroying your ST. Therefore, do not attempt the modifications unless you are

ready to assume all risks involved, understand the instructions, and have the appropriate tools, materials and skills required to perform the modification. Be sure to read the following instructions before you decide to begin the modification.

SUGGESTED MATERIALS
(38)

Two RCA Phono Jacks - Radio Shack ±74-346
One Submini SPST Toggle Switch - Radio Shack ±75-645
Six inches of miniature Shielded Cable - Radio Shack ±78-572
Ten inches of of 22 gauge wire.
Low wattage (40w), grounded tip soldering iron.
Long nose pliers. Diagonal Wire Cutters.
Wire stripper. Phillips screwdriver.

PROCEDURE

1) Disconnect everything plugged into your 520ST.
2) Remove the six screws from the bottom of the case. Notice the three screws from the rear of the case are longer than the three from the front. Be careful to keep these separate to avoid damaging the case when you reassemble your ST.
3) Gently lift the keyboard and disconnect the connector from the circuit board. Mark the top of the connector with a felt tip pen so you know how to plug the keyboard back in when you are done.
4) Remove the top metal shield surrounding the circuit board. Do this by removing the three screws in the front of the case, and straightening the tabs holding the shield in place.
5) Locate the Monitor Connector on the back panel of the ST.
6) Carefully drill two holes and mount the RCA phono jacks. I found room for mine between the Monitor connector and RF modulator, near the top of the case.
7) Drill another hole of the appropriate size for the switch.
8) Mount the plugs and switch to the back panel.
9) Cut and strip a length of shielded cable long enough to reach from a phono jack to pin one on the monitor connector. Pin one will be the top left pin as observed from the front of the ST.
10) Solder the center wire of the shielded cable to the center pin of one of the RCA jacks. This will be your audio output jack. Solder the shield to the tab on the RCA jack.
11) Solder the other end of the center wire to pin one on the Monitor Connector. Do not connect the shield on this end and make sure the shield doesn't short anything out.
12) Repeat steps 10 and 11 connecting the other RCA jack to pin two on the Monitor Connector. This will be the composite video output jack.
13) Solder a wire to the ground plane surrounding the outside of the circuit board. Connect the other end of this wire to the two tabs on the RCA jack you previously attached the shields to.
14) Carefully cut the Monitor Connector pin/conductor runing from pin 4 to the circuit board where it makes it right angle bend. This is the Monochrome Detect signal.
15) Separate the cut pin/conductor so they don't touch, and solder a wire from the top pin to one side of the SPST switch. Then solder a wire from the bottom pin to the other side of the SPST switch.
16) Carefully reassemble your computer in the reverse order you took it apart. Make sure you use the long case screws for the rear holes.
17) Reconnect everything plus your composite monitor. Hopefully, all went well and you can now select between color and monochrome modes by changing the switch you added.

AVATEX CABLE
(reprint: Huntsville Atari Users Group, Dec., 1986)

If you have an ST and an Avatex modem, the following cable will give trouble free telecommunicating. Get two DB-25 connectors and a 4 conductor cable and connect both ends as follows.

Pin 2 — Transmitted Data (TD)
Pin 3 — Received Data (RD)
Pin 7 — Signal Ground
Pin 8 — Carrier Detect

On the ST end ONLY, connect these two pins together as follows
Pin 4 — Request to Send (RTS)
Pin 5 — Clear to Send (CTS)

On the MODEM end of the cable ONLY, connect these two pins together as follows.
Pin 6 — Data Set Ready (DSR)
Pin 20 — Data Terminal Ready (DTR)

The ST handshakes using RTS/CTS, while the Avatex uses DSR/DTR. The software takes care of the handshaking when using this cable.

ATARI'S SMALL MIRACLES
Show Off Your Atari's Graphics

by Mark A. Brown

Welcome back and Happy New Year from the column for lazy programmers (which is, from my experience, a redundancy). These quick & dirty programs can teach you new techniques, show you practical examples of programming, but above all they're just for fun.

This month's column is once again dedicated to an aspect of the Atari 8-bit computers people love to show off: graphics. These five quick programs give you still more examples of how the Atari computer is better than any other 8-bit home computer in its specialty. All the programs are the type in, RUN, sit back, relax, and enjoy type. There is no interaction in the program itself, you just watch 'em work. Hope you enjoy them!

ECHO

The first two programs this month are from Carlos Moctezuma of Alexandria, Virginia. ECHO is just a pretty program that really doesn't have any new techniques or algorithms, but it is in the new combination of tried and true ideas that ECHO comes through. Enjoy the effect!

```
10 GRAPHICS 7+16:Z=1
20 FOR C=1 TO 8:SETCOLOR 4,C,8
30 READ N:SOUND 0,N,10,15
40 FOR P=1 TO 15:NEXT P:SOUND 0,0,0,0
50 A=INT(80*RND(0)+1):B=INT(156*RND(0)
+1)
60 COLOR Z:Z=Z+1
70 DRAWTO B,A:IF Z=60 THEN RUN
80 FOR L=15 TO 0 STEP -0.5:SOUND 0,N,1
0,L
90 NEXT L:NEXT C:RESTORE :GOTO 20
100 DATA 91,121,72,64,121,81,60,121
```

VISIONS

VISIONS is, again, a pretty program with no real new ideas. But that hardly reduces its worth. It is a spectacular example of the Atari graphics 9 mode (one color with in sixteen levels of brightness at once), showing the shading effects possible. Try changing the 79 in the FOR statement in line 10 to a 39 for a slightly different effect, or just fool around with the plots and drawtos -- rearrange them, change the numbers, etc. See what visions of your own you can come up with!

```
10 GRAPHICS 9:FOR I=1 TO 79
20 C=C+1:IF C)15 THEN C=1
30 SETCOLOR 4,C,0:COLOR I/2.5
40 PLOT I,X:DRAWTO 79-I,X
50 PLOT I,190-X:DRAWTO 79-I,190-X
60 PLOT 79-I,X:DRAWTO I,190-X
70 PLOT 79-I,190-X:DRAWTO I,X
80 X=X+1:IF X)190 THEN X=0
90 NEXT I
100 FOR D=1 TO 700:NEXT D:GOTO 10
```

ONELINE1 and ONELINE2

These two programs were an exercise of mine to see how much I could fit on to one line of code. Obviously, one line does not a commercial program make, and you can't do a whole lot in less than 120 characters (even with statement abbreviations, which I used extensively). But some decent effects came out nevertheless -- both from using tricks of the Atari (the artifacting of colors of high resolution pixels, and the weird sounds that come out when you give a SOUND statement funny volume numbers!)

```
10 GR.24:POKE710,0:POKE709,15:F.X=0TO3
9:FORY=0TO191:POKEPEEK(88)+256*PEEK(89
)+40*Y+X,X+Y:N.Y:N.X:POKE87,0
```

```
0 GR.18:POS.3,4:?#6;"ATARI'S SMALL":PO
S.6,8:?#6;"MIRACLES":FORA=0TO255STEP0.
5:SOUND0,0,0,A:POKE708,A:N.A:G.0
```

MIRROR

Finally, the most complex (and least spectacular) program of this month. MIRROR simply draws lines on the top half of the screen that are mirrored on the bottom half. This was not as simple as it sounds, because instead of simply plotting the points twice, the display list was altered to mirror the data.

A quick lesson on display lists is in order, although a better and much more comprehensive lesson can be found elsewhere. A display list tells the computer how to display memory on the screen. It is a list (located by the pointer at 560 and 561) of "statements" that tell the computer how each line on the screen is to be displayed. Try this short program:

```
GRAPHICS 0:FOR A=0 TO 19:PRINT PEEK(PE
EK(560)+256*PEEK(561)),:NEXT A
```

This will print the display list of the graphics 0 (all text) mode. The three 112's beginning the list tell the computer to skip 24 blank lines so data won't be off the top of the screen. The next number is a combination: subtract 64 and you get 2, the number that tells the computer it wants a line of text. The 64 added on to it says that the next two bytes are a pointer to the screen data. And indeed, the next two bytes are a pointer and, for this discussion, are meaningless numbers. Each 2 following the address is another text line, and if you look at the whole display list (change the 19 to 30 or so in the one line program above) you'll count 23 of them: add that to the 86 before the address and you have 24 lines of text.

What MIRROR does is take mode 11 (in the above discussion, the 2 is our mode, a text line -- mode 11 is a one color line twenty bytes wide, or graphics 8) on each line and add 64 to it to make each line point to it's memory. For the top half of the screen this isn't really necessary, but when we get to the bottom half of the screen, we load THE EXACT SAME DATA into the lines, but in reverse order. So whatever is drawn in the top half of the screen will be repeated in the bottom half; the data is read on to the screen twice.

So after all that, it's kind of a let down just to draw lines and have them mirrored. But the technique is there; see if you can find a good use of it. Synapse's game Encounter uses it to the extreme, making all that fast motion possible because it only has to worry about half the screen. Let me know of any really good effects you can come up with, and enjoy it!

```
10 GRAPHICS 22:DIM A$(1000):FOR X=1 TO
 3:A$(X,X)=CHR$(112):NEXT X:Z=PEEK(88)
+256*PEEK(89):Y=0
20 FOR X=Z TO Z+960 STEP 20:B=INT(X/25
6):A=X-256*B:A$(Y*3+4)=CHR$(75):A$(Y*3
+5)=CHR$(A):A$(Y*3+6)=CHR$(B):Y=Y+1
30 NEXT X:FOR X=Z+960 TO Z STEP -20:B=
INT(X/256):A=X-256*B:A$(Y*3+4)=CHR$(75
):A$(Y*3+5)=CHR$(A)
40 A$(Y*3+6)=CHR$(B):Y=Y+1:NEXT X:A$(L
EN(A$)+1)=CHR$(65):B=INT(ADR(A$)/256):
A=ADR(A$)-256*INT(ADR(A$)/256)
50 A$(299)=CHR$(A):A$(300)=CHR$(B):POK
E 560,A:POKE 561,B:POKE 708,15
60 COLOR RND(0):DRAWTO 159*RND(0),47*R
ND(0):GOTO 60
```

Atari's Small Miracles is always looking for good programs that are under ten lines, and if you have any you are especially proud of, send them to:

Atari's Small Miracles
c/o Mark A. Brown
7097 Game Lord Drive
Springfield, VA 22153

And if the programs are good enough, you'll see you name and program in this column!

I'd like to note that I can type, so you do not have to send a disk unless you have an incredible number of programs. If you do send a disk (for either giving me programs or requesting a disk copy of one already published) please make it a single density disk, because otherwise it'll have to wait until I have access to a double density disk drive.

I'll see you next month!

was much less expensive than springing for a BIT-3 or Austin Franklin board... and STILL is.) One source of dissatisfaction was the inavailability of commercially produced programs that utilized our OMNIVIEWs.

CDY Consulting, makers of the OMNIVIEW have recently introduced a new program that will produce an 80 column display...in a VT100 emulation mode. It's called OmniCom, and its author credits the public domain VT100 program as an early model for OmniCom. It has very many differences from that very basic program. First and probably most importantly, its use is restricted to OmniView owners. Second, it offers what the public domain VT100 program doesn't: file transfer capability.

OmniCom offers two modes of file transfer: the ever-popular Christenson protocol (Xmodem to most of us) in a checksum mode (as opposed to the CRC mode), and Kermit (Celtic for "free"). The only other program for the 8-bit machines that I know of that uses both is the latest version of John Palevitch's Chameleon program (but it's 40 columns:..). I tried both methods using this program and found probably the greatest (in my opinion) shortcoming of the program. Yes, both methods work. I highly recommend using a RAMDISK for uploading and downloading, because OmniCom apparently uses only a one-sector buffer for both transmission and reception in the Xmodem mode. I know this, because when uploading a file of approximately 40 sectors to a BBS, I had approximately 40 disk accesses. To my PHYSICAL disk drive. (My drive head still hates me.) In the Kermit mode, life is a little better. I estimate Kermit uses about a four-sector buffer. Or so.

Other features that are available: you can alter the screen and character color and intensity, change the aux1 and aux2 parameters in the XIO 34,36, and 38 calls to set up the RS232 port (which means you can configure OmniCom to work with almost ANYTHING!). The VT100 emulator also supports the PROFS protocol converter (mentioned in last month's review). There is also a software switch to switch between ASCII and ATASCII, so you get the standard Atari graphics when you log onto an Atari BBS. OmniCom also allows you to save these configurations to disk. To top it off, the user can completely redefine the keyboard to produce either characters or macro strings. That is, you can define CNTL-Z to be a clear screen character (for example), or "LOGON 1330DEON2" . And you can insert EOL's within the string. Finally, you

can dial your 835 or 1030 modem from the DIAL menu (41) selection.

So what's the bottom line? If you have a ramdisk, an OmniView enhancement, and $12 ($10 plus shipping), you can have a decent terminal program that does Xmodem, Kermit in ASCII or ATASCII, and emulates a VT100 (which is handy if you talk to mainframes), and, best of all, does it all in 80 COLUMNS. Do I recommend that you buy this program? You bet.

Exploring SpartaDos
Part 1 : Disk Architecture

Steven Novater, H.A.C.E

This is the first in a series of three articles dealing with that powerhouse disk operating system: SpartaDos. This first article will explain how SpartaDos organizes information on a diskette, and show how to access important disk parameters using the ACTION! language from OSS. In the next article we will develop a disk sector editor (in ACTION!). In the final installment, we will recover files from a SpartaDos diskette with damaged or missing directory sectors.

First, a little history of SpartaDos. In 1984, ICD Inc., a company born out of Spartan Software (makers of the Archiver chip for 810 disk drives) introduced the US Doubler for 1050 disk drives. They threw in a DOS with the doubler, SpartaDos 1.1 . There were some problems in compatibility with the almost universal standard set by Atari with DOS 2.0 . The future versions were designed to work with just the XL/XE series of computers, utilizing the extra memory in these machines to give both a higher level of compatibility and expandability that made XL/XE owners seriously consider "making the change."

SpartaDos is an "intelligent" DOS. It can accept almost any configuration of drives you can assemble. Got an ATR8000 and four 8" drives? Sparta can handle it. How about a hard disk? Sparta can handle it. In addition to being "drive smart", Sparta is a command processor type DOS, in the spirit of CP/M. As such, new commands can be added at will. Batch processing is also available, adding to the overall power of the system. Finally, SpartaDos is the only 8-bit DOS that uses a time/date stamp on your files. This may either be accomplished using a software clock that is set each time you boot, or the ICD R-Time 8 real-time clock cartridge.

Since most disk utilities were written for the

Atari machines with the minimal configuration in mind (one or two single sided single/double density disk drives), we notice that there is no commercial product available that will read or write past sector 720. So, if you have a double-sided/ double density drive, half your disk is inaccessible to these utilities (such as DisKey, Disk Wizard II, and so on).

In addition, much is generally known about Atari DOS 2.0. Read sector 360(decimal) and you've found the directory for the diskette. Each byte in the directory has a meaning that has been documented and re-documented by countless articles written over the past five years.

But what about SpartaDos? This one deviates from all the 2.0 type DOSes. If we want to read a directory sector, where do we look? What about these wonderful sub-directories? Let's take a look.

The SpartaDos manual for the SpartaDos Construction Set identifies four types of sectors on a Sparta diskette: Boot sectors, bit maps, sector maps, and data sectors.

BOOT SECTORS

In SpartaDos, the first three sectors of ANY density diskette are single density. These sectors contain the program that loads in the DOS, links it into the system, and enters the command processor.[*] In particular, the first sector contains a large table of important pointers to disk organization.

Offset (Decimal)	Description
9	first sector map of the MAIN directory
11	total number of sectors on diskette
13	number of free sectors on diskette
15	number of bit map sectors
16	first sector number of first bit map sector
18	sector number to begin the file data sector allocation search.
20	sector number to begin the directory data sector allocation search
22	diskette volume name (8 bytes)
30	number of tracks on diskette (note the most significant bit is set if the drive is double sided)
31	size of sectors on diskette. 0 means 256 byte sectors, 1 means 128 byte sectors.
32	major revision number of the DOS on this diskette
33	number of buffers reserved for sector storage (not for version 2.x or higher diskettes)
34	default drive if this diskette is booted

35 | reserved
36 | reserved
37 | number of sectors in main DOS boot (version 1.x only)
38 | volume sequence number (2.x only)
39 | volume random number (2.x only)
40 | first sector map of file specified by BOOT command (2.x only)
42 | write lock flag (2.x only)

Table 1
Sector 1 Vital Information

SECTOR MAPS

SpartaDos disk structure may be thought of as consisting of many sector maps. There are sector maps for the directory(s), sector maps for the files, and sector maps for the sector maps. This allows considerable expansion. How do these sector maps work? Take the directory sector map for example. This is a two byte value (high,low) at offsets 9 and 10 in the first sector on the diskette (see Table 1). This value is the sector number that contains the list of sectors used in the MAIN directory. If, for example, this value were 36, we could read sector 36 and see a list of the sectors that contain the directory.

When reading a sector map, the first two bytes point to the previous sector map, while the next two point to the next sector map. The first two values are zero if this is the last sector in the sector map, while the second two values are zero if this is the last sector in the sector map. By reading these values, we can step our way through the sectors in a sector map. This is analogous in some respects to the method Atari DOS 2.0 uses in linking sequential files, except Atari DOS 2.0 stores this information at the end of a data sector.

After the first four bytes in the sector map, the rest of the sector consists of a list of two byte values , each representing a sector number. In the case of the directory sector map, this list is a list of all the sectors comprising the directory. In the case of a file sector map, this would contain the sector numbers of all the sectors occupied by that particular file.

BIT MAPS

A bit map on a Sparta disk resembles the VTOC sector on an Atari DOS 2.0 diskette. That is, it is a map of used and unused sectors on a diskette. In a bit map, each byte represents 8 sectors, with the seventh bit as the first sector in a group, and bit

zero represents the eighth sector in the group. It is possible to have more than one bit map sector on a diskette (Simple arithmetic - divide the number of sectors on a diskette by 8 and you have the number of bytes needed for a bit map. Then divide this by the number of bytes in a sector to find the number of bit map sectors.) If there is more than one bit map sector on a disk, then the bit map sectors will be sequential on the diskette.

DIRECTORY DATA STRUCTURE

In order for SpartaDos to find a file on a diskette, the directory must be accessed. Unlike Atari DOS 2.0, this is generally NOT in a fixed location on a disk. Offsets 9 and 10 in the sector 1 deliver the sector number of the first sector map of the MAIN directory.

Each entry in the directory is 23 bytes in length. These 23 bytes contain the filename, filetype, date/time , length, first sector map number, and the status of the entry. The first entry in a directory is special. It contains the first sector map of the parent directory (offset 1 and offset 2 - a zero indicates that this is the MAIN directory), the length of the directory (3 bytes at offset 3), and the directory name (8 bytes at offset 6). The parent directory entry for a subdirectory contains everything (date/time, name,first subdirectory sector map) except the length. Each 23 byte entry (the first "special" entry is also 23 bytes) contains the information in Table 2.

---

offset	description
0	file status byte. zero indicates end of directory file

The following bits when set have the meaning:

```
        bit 0 - entry is protected
        bit 3 - entry is in use
        bit 4 - entry has been deleted
        bit 5 - the entry is a subdirectory.
1       (2 bytes) first sector map of the file
3       (3 bytes) length of the file
6       (8 bytes - space padded) filename
14      (3 bytes - space padded) file extension
17      (3 bytes) date file created (dd/mm/yy)
20      (3 bytes) time file was created
(hh/mm/ss).
```

Table 2
Directory entry structure

DATA SECTORS

Data sectors contain nothing but data. 256 bytes if you're running double density, and 128 bytes if you're running single density.

Beginning to get the idea? We can read the first sector on a diskette, and tell what kind of diskette we have, what SpartaDos version created it, where the directory is, where each file map starts, how long each file is, and all sorts of other data useful to the SpartaDos hacker.

There is a command in SpartaDos versions 2.x and higher that gives some of this information. It's called CHKDSK. This internal command is a call to CIO with iccom = 47, and yields the volume name, random and sequence numbers, sector size, formatted bytes on diskette, available bytes on disk, and write lock status. To get at the rest of the information in the table in sector 1, we must do a little more work (play?).

I chose to implement this in ACTION!, as my first "real" programming project in that language, serving the dual purposes of getting the information from sector 1, and learning ACTION!. The following listing (available upon request in file form) will read the first sector, and translate the info (bytes) into a more-easily read form (English). We start by defining the SIO procedure, then set up a more general procedure to read a sector into a memory buffer, then this buffer is read and translated to the screen (or printer, if you set the device byte to "P:" in the main procedure).

Next month we will read a directory map sector, and access the directory information DIRECTLY, without benefit of the SpartaDos command processor. Until then, keep hacking.

(This article was written with heavy borrowing from Chapter 19 of the SpartaDos Construction Set, ICD, Inc.)

```
PROC SIO=$E453()

MODULE

BYTE ARRAY Buffer_in(256)=$8000
BYTE ARRAY Buffer_out(256)=$80FF
BYTE Ddevic=$300,  ; device serial bus id D1-D4
($31-$34)
    Dunit=$301,     ; disk or device #
    Dcomnd=$302,    ; operation command code
$52=get sector $50=put sector
    Dstats=$303,    ; status & operation info
$40=read $80=write
```

FEBRUARY 1987

6   HACE NEWSLETTER
     FEBRUARY 1987

```
    Dbuflo=$304,   ; buffer low address
    Dbufhi=$305,   ; buffer hi address
    Dbytlo=$308,   ; bytes to be transferred low
address
    Dbythi=$309,   ; bytes to be transferred hi
address
    Dauxl=$30A,    ; sector address low to be
read/written
    Daux2=$30B     ; sector address hi to be
read/written

BYTE dev,cmd,aux1,aux2


;********************************
PROC Read_Sector(CARD Sector)

Ddevic=$32        ;$31 FOR D1:,$32 FOR D2:, ETC.
Dunit = 2         ;DRIVE NUMBER 2
Dcomnd=$52        ; read command
Dstats=$40        ; set 2 way status to a read
Dbuflo=$00        ; setting buffer address lo
Dbufhi=$80        ; setting buffer address hi
Dbytlo=$FF        ; set number of bytes to be
transfered to 128
Dbythi=$00


Daux2=Sector/256
Daux1=Sector-Daux2*256

SIO()             ; call serial I/O routine

IF Dstats=1 THEN RETURN FI

  PutE()
  Print("Error in read sector.  Error code=")
  PrintBE(Dstats)
  RETURN


CARD FUNC conv(BYTE I)
RETURN(Buffer_In(I)+256*Buffer_In(I+1))

CARD FUNC rconv(BYTE I)
RETURN(Buffer_In(I+1)+256*Buffer_In(I))


PROC MAIN()
CARD J,I,secmap,numsecs,freesecs,
     firbit,datalloc,diralloc,bootmap
BYTE ARRAY VOL(8)
BYTE bitsecs,notrks,sizsec,dosrev,
     numbuf,default,dosboot,volseq,
     volrand,wrtlock
```

```
;parameters from SpartaDos
   Read_Sector(1)

;first do the 16 bit values

secmap  = conv(9)
numsecs = conv(11)
freesecs= conv(13)
firbit  = conv(16)
datalloc= conv(18)
diralloc= conv(20)
bootmap = conv(40)

;now for the 8 bit values

bitsecs = Buffer_In(15)
notrks  = Buffer_In(30)
sizsec  = Buffer_In(31)
dosrev  = Buffer_In(32)
numbuf  = Buffer_In(33)
default = Buffer_In(34)
dosboot = Buffer_In(37)
volseq  = Buffer_In(38)
volrand = Buffer_In(39)
wrtlock = Buffer_In(40)

; now print the volume name

PrintE("SpartaDos Disk Descriptor ....")
PrintE()

PrintE("Disk has volume name ")

FOR I= 22 TO 29
DO
   PrintF(" %C ",Buffer_In(I))
OD
PrintE()

Print("First sector map of MAIN directory: ")
PrintF(" %U %E %E",secmap)

Print("Total number of sector on diskette: ")
PrintF(" %U %E",numsecs)

Print("Number of free sectors on diskette")
PrintF(" %U %E",freesecs)

Print("Number of bit map sectors used")
PrintF(" %I %E",bitsecs)

Print("First bit map sector: ")
PrintF(" %U %E",firbit)
```

7   HACE NEWSLETTER
     FEBRUARY 1987

```
Print("Sector number to begin file data sector
allocation search: ")
PrintF(" %U %E",datalloc)

Print("Sector number to begin directory data sector
allocation search: ")
PrintF(" %U %E %E ",diralloc)

IF (notrks&$480) THEN
   PrintE("Disk is double sided ")    ;NOTE! the
high bit is set if the drive is double sided
ELSE
   PrintE("Disk is single sided ")    ;NOTE! the
high bit is set if the drive is double sided
FI

notrks = notrks-(notrks&$480)

;NOTE! the high bit is set if the drive is double
sided

Print("Number of tracks: ")
PrintF(" %I %E",notrks)

Print( "Sector size  : ")

IF (sizsec=0) THEN
   PrintE(" 256 bytes ")
ELSE
   PrintE(" 128 bytes ")
FI

Print("DOS revision number: ")
PrintF(" %I %E",dosrev)

Print("Default drive : ")
PrintF(" %I %E",default)

; the next is applicable only to version 1.x
diskettes

IF (dosrev<20) THEN
   Print("Number of buffers to reserve: ")
   PrintF(" %I %E",numbuf)
   Print("Number of sectors in DOS boot: ")
   PrintF(" %I %E",dosboot)
FI

IF (dosrev>20) THEN
   Print("Volume sequence number of diskette: ")
   PrintF(" %I %E",volseq)
```

```
Print("Volume random number: ")
PrintF(" %I %E",volrand)

Print("First sector map of file specified by  the
BOOT command: ")
PrintF(" %U %E",bootmap)

Print( "Write lock flag: ")
PrintF(" %I %E",volrand)
PrintE("    $FF - diskette is locked")
PrintE("    $ 0 - diskette is not locked")
FI

RETURN
```

ATARI 400 ALARM SYSTEM
by Jerry Sullivan

This doc file explains how to set up a burglar alarm using an Atari 400 and (optionally) a program recorder. Having both of these sitting around gathering dust I decided to try to put them to some use utilizing the input and output capacity of the joystick ports. All parts were found at Radio Shack and are listed at the end of this article.
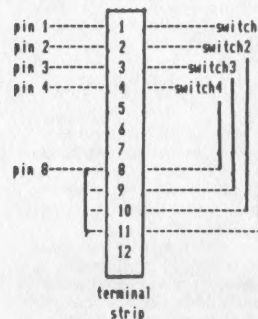
The first step is to wire joystick port 1 for input. Buy a joystick extension cord and cut it in half (the half you don't use here is perfect for half of a 850 modem cable). Using an ohmmeter you can determine the pin outs and wire colors are as follows:

Pin #	Wire Color
1	Brown
2	Red
3	Orange
4	Yellow
5	Green
6	Blue
7	Black
8	Gray
9	White

For the input part of the alarm we will use pins 1,2,3 and 4 for one side of the switches and pin 8 for the ground side of the switches. Use a twelve position insulated terminal strip and connect the magnetic contact switches run from the doors you wish to monitor to it as is shown in the schematic (Figure 1). Then connect the wires of the joystick extension cable as shown. For test purposes you can connect the switches directly to

the terminal strip and wire them to the doors later. The program I wrote for this will only work properly using 1 NO (normally open) switch (Pin #1) and 3 NC (normally closed) switches (Pins #2,3,4). To use other switches some minor reprogramming of the program will be required.

Figure 1


```
pin 1-------|  1  |-------switch1
pin 2-------|  2  |-------switch2
pin 3-------|  3  |-------switch3
pin 4-------|  4  |-------switch4
            |  5  |
            |  6  |
            |  7  |
pin 8-------|  8  |
            |  9  |
            | 10  |
            | 11  |
            | 12  |
          terminal
           strip
```

When you have this wired together enter and run this short program:

```
10 ? PEEK(632):FOR X=1 TO 300:NEXT X:?:GOTO 10
```

PEEKing 632 will return the value coming back from joystick ports 1 and 2. This will give you the number you will be receiving when all doors are closed (if your switches are wired as shown and are in their closed positions). You can then 'open' each switch and write down what number is returned when each door and each combination of doors is open. There will be a total of 16 ranging from 0 to 15. One of these will be normal status and the other 15 will be when one or more doors is open. If you use pin 1 for a NO switch and pins 2,3 and 4 for NC switches you do not need to do this as the program is already set for that combination.

Now that we have the computer recognizing when a switch is open we will have to have it tell us. There are several ways to do this. I have written the program to output to 4 indicator lights as well as the internal bell of the computer or to a tape in the cassette recorder through a tv speaker. I opted to use joystick port 3 for the output to the indicator lights. This requires another twelve position terminal strip, joystick extension cable, indicator lamps, 5volt DC power supply, 8 2.2K

resistors , a breadboard, a QUAD AND gate (RS# 276-1822) a little wire and some soldering. See Figures 2(A-D) for the schematic for properly wiring up this combination. The figure in the middle with 14 pins (7408) is the Radio Shack Quad And gate, the R represents a 2.2K resistor and the L represents the indicator lamps. I separated it into four schematics for purposes of clarity. As you will notice pin 7 of the 7408 is connected to the negative side of the external power supply and pin 8 of the joystick port one time only. Also the 7408's pin 14 is connected to the plus side of the 5vDC external power supply only once. All else is duplicated for each pin. I tied two inputs of each and gate together, if you can find a 7407 QUAD AND gate you will only have one pin in place of the two. Apparently Radio Shack does not make this one, which is made by Texas Instruments.
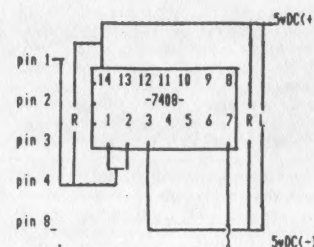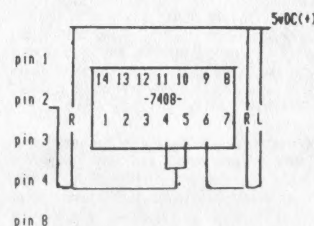
Figure 2A


```
pin 1
      | 14 13 12 11 10  9  8 |
pin 2 |        -7408-        |
      | R   1  2  3  4  5  6  7   R  L |
pin 3

pin 4

pin 8
```

Figure 2B


```
pin 1
      | 14 13 12 11 10  9  8 |
pin 2 |        -7408-        |
      | R   1  2  3  4  5  6  7   R  L |
pin 3

pin 4

pin 8
```
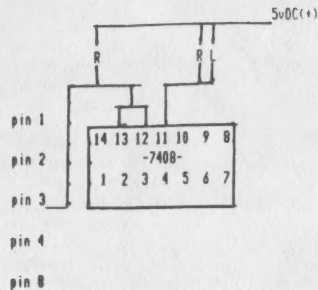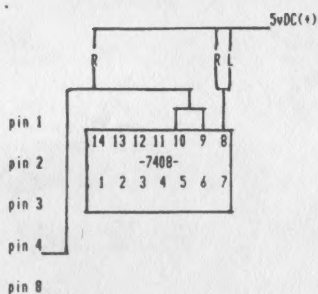
Figure 2C

Figure 2D

pin 8

'Firing' of the pins 1-4 is accomplished by first POKEing 54019,48 which turns joystick ports 3 and 4 into output mode then POKEing 54017 with the correct number to fire any combination of pins you desire. This is covered much better in 'Mapping the Atari' by COMPUTE! BOOKS. The program listing has the correct numbers for port 3 incorporated in it and you can experiment with it by running this short program:

```
5 POKE 54019,48
10 FOR X=241 TO 255:POKE 54017,X
20 FOR DELAY=0 TO 50:NEXT DELAY:NEXT X:GOTO 10
```

Alright now we have a working burglar alarm. ㊼ Just plug in the external power supply run the ALARM400 program and your off and running. The program will prompt you to answer the following: (M)usic or (B)ell or (N)one

If you answer M or m the program will attempt to start the cassette motor on an alarm state. So if you put a tape in and have the play button down the tape will start playing ,through the tv speaker, when a door is open. Lights will also light to indicate the source of the intrusion.

If you answer B or b the program will ring the internal bell of the computer as well as light the lights.

If you answer N or n the program will only display the lights.

To reset the program after a door is opened you can (BREAK) and type G.100. This will reset the alarms (assuming the switches are reclosed) and restart the program. Lines 41-44 should be changed to indicate the location of your particular switches.

Now you should have enough to start on to build your own system. There are many possibilities available. I use the 5 volts DC to light the indicator lamps but it could just as easily operate a buzzer, a relay to open and close 120 volts AC or 12 volts DC which could operate lights, sirens, your coffee pot etc, etc. With a little more programming you could add a clock and use it as an alarm clock or a timer to turn lights and appliances on and off. Not being an electrician or instrument tech I really do not fully understand what I have done here and expect that there are easier more efficient ways of doing this, but the bottom line is it does work.

I have only used half of the inputs and outputs available. Port 2 could easily be hooked up as input and port 4 for output applying the same techniques used here.

REFERENCES

Semiconductor Reference Guide 1985 Edition 21984 by Radio Shack. (p.3,37).

Mapping the Atari by Ian Chadwick 21985 by COMPUTE! Publications. (p.54,130-132).

Creative Computing April 1983 Outpost:Atari article.

A.N.A.L.O.G. July 1986 (supposed to have a relevant

ACCENT ON BASIC COMPUTING
A Brief Introduction to Spreadsheets
㊽

by Ron Peters

Last time we explored word processing, and how it replaces your typewriter, scissors, paste, wastebasket and Xerox machine. Now let's take a look at the so-called "spreadsheet" programs and see what they can do for us.

Perhaps some of you have seen the traditional, green accounting ledger paper, with several columns across the top and 25-30 lines running top to bottom. The "bean counter" types use these ledger sheets for budgeting, inventory, and ledger purposes. A typical budget sheet would have the months of the year across the top, and line items (advertising, telephone, salaries, travel, etc.) running down the left side.

Then someone plugs in the dollar amounts for each item under every month, like the following (brief) example for a company we'll call Sicktemps (they supply sick temporaries to companies with lower than average absenteeism problems):

```
1987 EXPENSE BUDGET - SICKTEMPS, Inc.
Item        Jan  Feb  Mar  Apr  May  Jun  Total
Insurance    50   50   50   50   50   50    300
Salaries    100  100  100  100  100  100    600
Supplies     25   25   25   25   25   25    150
Telephone    10   10   10   10   10   10     60
Totals      185  185  185  185  185  185   1110
```

The above could easily be a home budget, or a listing of inventory items in different categories, or names and addresses, etc. In other words, the "spreadsheet" does not have to be just numbers, it can contain names, lists, text — anything.

OK, back to the task at hand. Looking at the above spreadsheet it is obvious that the columns (down) and rows (across) are cross-totaled. Thus, you can easily see the total expense for each line item (e.g., Insurance) and/or each month, with a grand total at the lower-right.

If you are doing this budget on paper, and decide that supplies are going to cost $35 per month, instead of the original $25, you will have to erase all the numbers in that row, as well as all the totals, and do a lot of addition to get a corrected budget sheet.

Here comes the electronic spreadsheet. Instead of putting these numbers on a sheet of paper, you "type" them onto a worksheet in your computer. And, you set up the worksheet so that it will automatically give you totals for each row and column. Now, change the supplies amount to $35 and the program instantly recalculates all the totals -- before you can blink!

This feature also gives you the "what if" option; that is, the ability to change any of the data to see the results on the total. For example, if you were doing a home budget, you could ask "what if" my wife were to spend $300 per month instead of $1000 per month on clothes? The final results would be immediately calculated (but not admissible in divorce court).

"Big deal," you say. Well, it is a big deal if you have 120 columns and 234 rows, with a lot of changes to make. A computer with 512K of memory can typically hold a spreadsheet that would be 20 feet wide by about 40 feet high on paper. Try working on that within the confines of your 10 X 10 foot office!

"How do I look at that big sucker on my computer screen?," you say (again). Through a series of "windows" that allow you to look at any portion of the spreadsheet on your computer screen.

Each portion of an electronic spreadsheet (called a cell) can contain numbers, text, or formulas. The formulas can range from simple sums of columns (for example) to complicated algebraic expressions.

Any portion (or all) of the spreadsheet can be printed or saved on a floppy disk for later use. You can even create a "template" spreadsheet, which is nothing more than a spreadsheet with the labels and formulas, but without the actual data. For example, our budget template would have the row and column labels, and the total formulas, but no actual monthly dollar amounts. Then we could use the budget template over and over for different year budgets.

Some spreadsheet programs (like *LOTUS 1-2-3* or *VIP Professional*) can produce graphs of the data contained in a spreadsheet. Or you can transfer any portion of the spreadsheet to a document in your word processor.

Perhaps the most practical application I have found for using a spreadsheet at home is for doing my income taxes. Using a tax template (that purchased from the NOVATARI disk library) I can plug in all the numbers for each tax schedule and see the final results (hopefully a refund). Using the "what if" function here can be real fun!

Like a word processing program, you don't have to spend a fortune to get a good spreadsheet program that will more than do the job for you. I use *Synfile+*, which costs about $40. I'm sure there are a lot of other programs out there in the same price range that are equally as good. Shop around.

— XE 12 XL —

TIPS'N'TRAPS
First Aid for Atari Adventurers

by Jim Stevenson Jr.

I recently received a letter from one of the readers of TnT, MaryLou White. She has donated a complete set of solutions to the popular Activision game Hacker. Hopefully, over the next few months, along with the standard Q&A format, we'll reveal what she sent us. Also, Merlin's Litterbox has pulled through as being a new TnT source. Call it at (703) 250-7303. Also call Electronic Age at (703) 620-0851, and ARMUDIC at (703) 569-8305. If you don't have a modem, call me at (703) 378-4093. And, now for some more adventure snags.

ERRATA: -- ULTIMA IV <--
A. The location of the Mandrake Root is located at D'G''-L'G'', not D'H''-L'H''...Sorry Bellkose. Also, when searching for either Mandrake or Nightshade during the two new moons, immediately start pressing "S" for "Search" several times very fast, so that you may find those ingredients about three times without waiting for another double new moon.

-Del Whetter
Eugene, OR.

HACKER
Q. I thought I was doing pretty well in Hacker until hitting a dead-end on the last lap. I can get all the pieces of the message, but I always run out of time on my way to deliver it to Washington. I don't think I've been dawdling, so how come I always get zapped? If I could get all the pieces without ever being spotted by the satellite, would it give me more time, or is there some other solution I've missed?

-Larry Franks

A. The secret to winning the game is not to beating the clock (which was also my original assumption). Also, being spotted by the satellites doesn't seem to have any effect. The real trick in winning is not to take more than a total of 173 steps from the start of the game to the finish at Washington D.C. If you make even one wrong step, it will add up to more than 173 steps so you must know the route and follow it carefully. You can take as much time as you like in playing the game. Speed is not a factor. Leave it to Activision to throw in a tricky ending.

-MaryLou White

PLANETFALL
Q. Is there food in the kitchen so that you don't starve? Also, are the goos in the survival kit simply food?

-"Red Baron"

A. There is food in the kitchen that you can put in the canteen. the goos in the survival kit are also food.
-"Dinty More"

Q. You mean you're supposed to put the food in the canteen? Do you ever get thirsty in the game?
-"Red Baron"

A. The food in the dispenser in the kitchen also quenches thirst (I think).
-"Dinty More"

ZORK II
Q. I get stuck in places like the oddly angled room, where the cakes are, and pushing the buttons, I don't know which button does what.
-"Wolverine"

A. The cakes are in the tea room, or something like that. One of the cakes makes you big, one makes you small, and one makes you explode. Experiment. The buttons control the carousel. One makes it go faster, one slower, and the other makes it stop. The oddly angled rooms have something to do with baseball. You have to run the "bases". Have you seen the "bat"?
-Bill Mehojah

Q. How do you get the key from the unicorn?
-"Sci-Fi"

A. There are some caves (I guess you know where those are), and one contains a dragon. Theres a person being held prisoner by the dragon. So, you have to defeat the dragon in order to free that person in order for that person to get the key and give it to you.
-Bill Mehojah

ZORRO
Q. What do you do with the bottle, and how do you get the horseshoe?
-"Crackerjack Kid"

A. You put the bottle by the farthest person at the bar. Then you get on top of him and jump up and down. When you get up to the top there push another person off and they will get caught on the chandelier and the thing at the bottom will raise and you go down there and get the chalice. But make sure you opened the door where the ball is. (For details as to how to do this, see last month's Tips 'N' Traps--J.S.)
-"Hot Rod"

CHAMPIONSHIP LODE RUNNER
Q. I was wondering how you get past the first screen. I can't get passed it.
-"Hot Rod"

A. To get past the first screen, you must first get the

two bottom lodes. First keep the two men on the opposite from you as you break down the wall on your side (delaying a little after the first segment), then run up and across the staggered floor, drop and get the lode, and rush to the opened ladder before the wall closes. Trick the men to your side and proceed to other side to repeat the operation there and get the second lode. Then get the men to come across the ladder/wire level and trick them into falling onto the lower level. From there they can't escape, and you can finish the screen without problems. Believe me, though, subsequent screens (particularly no. 9) are just as tricky, but that's what makes it so much fun!
-"Carlos Moctezuma"

OGRE
Q. What are you supposed to do to stop that tank?
-"Hot Rod"

A. When you set up the play field, you place armor and infantry divisions. In playing the game, the "menu" at the right changes from "defensive movement" to "attack" etc., during attack, fire artillery, etc. at the ogre.
-"Bellkose"

HITCHHIKER'S GUIDE TO THE GALAXY
Q. I can't put the fluff together, so I just quit. Any ideas?
-"The Distorted"

A. You need to get the flowerpot from inside the whale, but to do that, you must get real tea to use as Brownean Motion. Also, DO NOT finish the War Chamber sequence until AFTER getting the flowerpot! Kill yourself if you show up there before getting the pot.
-"Nino Greasmanelli"

Q. After you wake up and get the babel fish in your ear, what word do you listen for in the Captain's poems, then what do you do to avoid getting him mad about it, then how do you get off the ship?
-"Dagor"

A. The word to listen for varies. You type [FLIP SWITCH] when you're in the hold because there's that keyboard with a switch. It will tell you to open it, type in the nth word from the SECOND verse of the captain's favorite poetry. The number of the word is random (usually 1-3) and the word is also random. To get to see the second verse (you must be masochistic!) type [ENJOY POETRY] when he's reading the first verse. You will then go back to the hold. Don't forget your gown and towel! Type [TYPE "insert appropriate word here"] and the case will open. [GET PLOTTER]. When you get thrown into the airlock, you can just wait it out (which is kind of funny) or you can use your thumb.
-"Nino Greasmanelli"

Q. I've gotten all the stuff to win, but the Buggblatter keeps killing me. Can someone tell me how to get past it?
-"Buckaroo"

A. Have you carved your name into the stone? When you are at his place, you have to carve your name into the stone. It makes the creature think that it has already eaten you, so it goes off and takes a nap or something. It's been a long time since I've played, so I really don't remember on what order you have to do what, but I do recall something about wrapping the towel around your head so that the beast won't see you (since you can't see it, it can't see you). Then write your name on the stone with a sharp rock or something, and type in quotes your name (the characters name.)
-"Bellkose"

ULTIMA III
Q. Does any one know in witch dungeon and what level the mark of the snake is? That's the only mark I can't find.
-"Weird Al"

A. There is a dungeon on an island near the dungeon that is surrounded by lava. (I think its SE of B.'s castle). You can get there either by moon gate or boat. Go to the bottom level. Now, follow these directions to the letter, and they will take you to the rod (should take you). From the stairs, facing south, go 1 south, 3 west, 2 south, 2 east, 2 south, 6 west (2 "strange winds in there), 2 south, then east to the wall. Use a gem on level 8 and you will see what I mean.
-"Bellkose"

Q. Okay, I am ready to go kill Exodus. I have the marks, cards, etc., but I can't get past the snake, even though I have the mark of the snake.
-"Weird Al"

A. If you have all 4 marks, go to the front of the snake. Hit "Yell" and then type in "evocare", "delcare", or "" depending on the copy (I think) when it asks what to yell. If all goes well, you should be teleported to the other side of the snake.
-"Barracks Rat"

ULTIMA IV
Q. What is the path through the maze on level 6 of the abyss? I've tried several times to get through and can't do it! Frustrating! I'm so close to finishing!
-Pete Kilcullen

Q. Does anyone know where the magic wheel is? Also I need to know some details about sheppheards, what weapons can they use, do they ever get to use magic, etc.
-Andy Patton

A. I don't know the exact coordinates of the wheel, but it's in the ocean. Somewhere down off to the right from Lord B.'s castle. You'll see light blue water surrounding dark water or something like that. I think that's where the wheel is. Maybe that's where the horn is. Anyway, just search there.
-"Red Baron"

(51)

Q. Does the whirlpool go somewhere like in Ultima III?
-"Red Baron"

SANDS OF EGYPT
Q. How do you get into the pyramid, if you're supposed to get into it at all?
-"Red Baron"

A. You hook the scepter that you got at the top of the pyramid to the bottom of the pool on that plug. You pull on the scepter, unhook, and take the scepter down with you to the underground tunnel.
-Jim Stevenson

A. It's somewhere in the desert, and you have to kill the snake for it, and put it in your canteen. (You get the water for it later on in the game.) From the very start, type "N, N, N, W, W, GET SHOVEL, N, KILL SNAKE, WITH SHOVEL, W, W, W, GET CANTEEN, N, FILL CANTEEN (WITH OIL FROM DEAD SNAKE)
-Jim Stevenson

ALTERNATE REALITY
Q. Has anyone been able to join a guild yet? I have tried and I can't seem to join one.
-"Hot Rod"

A. Personally, I have not been able to join any guilds, but I know this guy who says he did. He told me that he killed an apprentice as he was leaving a guild then went back in and there was room for him to join. He also told me that it takes a lot of gold to join.
-"Buckaroo"

HARDHAT WILLIE
Q. Is there any way to kill that Pac-Man guy? Also, I got all the caterpillars and eggs, now what?
-"Out of Data"

A. Yeah, the whirlpool goes to Loch Lake. The candle of love or something like that is back there in a town.
-"Dinty More"

Q. What is the candle used for?
-"Red Baron"

BOUNTY BOB
Q. Does anyone know some shortcuts for jumping ahead to different levels of "Bounty Bob Strikes Back"? I already know how to jump to the 5th screen from the 1st screen.
-Bill Godfrey

- XE 20 ST -

---

(52)

(') and other Atari classics. A charming touch was the presence of local children, dressed in Atari warm-up suits, who were demonstrating the videogames for the benefit of us repressed, three-piece-suited adults. (Note: General attendance to CES is limited to those 18 and over. The children were there by special permission.) I walked up to a game playing on a 7800 system and idly picked up a joystick. An angelic blond 9-year-old boy, who should've been out playing with June Locknart and Lassie, picked up the other joystick and politely asked me if I would like to play against him. "Sure." I said, smiling indulgently. Five minutes later, the lad's razor-honed videogame reflexes had blown me away. My smile faded and I stomped off, grumbling.

Some final software news: Psion Chess will be distributed by Atari. No price has been set. Electronic Arts is putting the finishing touches on their Chessmaster 2000, a fantastic ST chess game they are porting over from — you guessed it — the Amiga. I had the opportunity to play it during an Electronic Arts party, and it was very colorful, very fun, and VERY difficult. Retail should be under $50. (A review of ST chess programs will be appearing in an upcoming issue of START.)

Thanks for bearing with us. We've tried, despite various technical gremlins, to give the Atari community as much information as quickly as possible. Among other features, we hope to present DIGITIZED PHOTOS on ANTIC ONLINE which you can view. Our goal is to become one of the fastest sources of Atari information (including visuals) short of a business report on CNN.

Again, thank you, and keep watching ANTIC ONLINE for programming tips, graphics demos, and late-breaking news reports from the Atari community.

Captured and uploaded by Sean Kelly 72305,1310, from Antic Online (Type GO ANTIC at almost any ! prompt on CompuServe).

2-87

Y-Modem
Excerpts from Ralph Walden

With Express 3.0 adding CRC error checking,and Osbis BBS software (SKYLINE 303-497-0270) adding CRC and the new Y-modem protocol,there have been alot of questions raised about the difference between the various downloading protocols .For a long time,Xmodem has been the standard way to transfer a file between your terminal program and a BBS. Xmodem sends 128 bytes of data at a time,plus 4 bytes of information your terminal program uses to see if the data came through correctly.(see a previous MILE HIGH MAGAZINE for my article on XMODEM error checking.) If there was some noise on the phone line,your terminal program will recognize it and ask the BBS to send the 128 byte block again. USUALLY this means an error free file,but XMODEM is not perfect and occasionally errors will slip through, and you have a bad program. The program may run fine until it gets to the garbled section and then crash. CRC is a more accurate method of transfering files.It sends five bytes of control information for every 128 bytes,and uses a sophisticated mathematical algorithm to check

for errors.It is almost impossible to get a bad file transfer with a CRC download. Y-Modem is an extension of CRC.Instead of sending 128 bytes at a time,it sends 1,024 bytes at a time.This makes for slightly faster file transfers for a direct call, and will halve the download time if your using a packet type long distance service like (PC-PURSUIT). On the other hand, if you have a bad phone connection, Y-Modem will be slower since each bad transfer will cause 1,024 bytes to be resent instead of 128 bytes. If your terminal program supports it, you should always use either CRC or Y-Modem. ST-Users should set their defaults to CRC. If your terminal supports Y-Modem, it will automatically switch to Y-Modem if the BBS software supports it also.

8-Bit users,if you don't have a copy of Express 3.0, you should get one. Skyline has it available all the time. So far only the 850 version is supporting the CRC, but rumor has it that the 1050 and Mpp versions will be held up till the author (Keith Ledbetter) finishes version 3.1 which will support not only CRC, but Y-Modem also. You ST'ERs allready have a nice implementation of Y-Modem in ST-Term, with rumored patches to FLASH in the works. Try it .YOU'LL LIKE IT!!!!

Dave Lloyd-Editor-